2017

Discrete event simulation for the analysis of artillery fired projectiles from shore

Sheatzley, James Garrick

Monterey, California: Naval Postgraduate School

http://hdl.handle.net/10945/55536

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

DISCRETE EVENT SIMULATION FOR THE ANALYSIS OF ARTILLERY FIRED PROJECTILES FROM SHORE
by
James Garrick Sheatzley
June 2017
Thesis Advisor: Arnold H. Buss
Second Reader: Susan M. Sanchez

This thesis was performed at the MOVES Institute.
Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	Form Approved OMB No. 0704–0188

1. AGENCY USE ONLY *(Leave Blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	June 2017	Master's Thesis 07-06-2015 to 06-16-2017

4. TITLE AND SUBTITLE
DISCRETE EVENT SIMULATION FOR THE ANALYSIS OF ARTILLERY FIRED PROJECTILES FROM SHORE

5. FUNDING NUMBERS

6. AUTHOR(S)
James Garrick Sheatzley

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
N/A

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
The views expressed in this document are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol Number: N/A.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release. Distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

The purpose of this thesis is to provide decision makers a tool for analyzing the effectiveness of current United States Marine Corps artillery systems conducting anti-access area denial operations. Artillery effects against land targets are documented and understood, but a knowledge gap exists regarding the effectiveness of artillery operations in the littorals. Expeditionary Fire Support Model–Maritime (EFSM) is a discrete event model that simulates current capabilities of Marine Corps artillery systems. Integrating an existing naval convoy model, two proof-of-concept littoral scenarios are presented that represent battalion and regimental artillery task organizations tasked to deny freedom of navigation (area denial) and stop an amphibious naval convoy (anti-access). Results from a designed experiment indicate artillery systems provide commanders a limited area denial capability, and should be employed where naval forces are limited in maneuverability and follow known routes close to shore. Overall, artillery achieve higher destruction rates in the area denial scenario than anti-access scenario. Factors important for successful anti-access area denial operations include unmanned aerial system speed and firing delay of the M777A2 lightweight howitzer. Data produced during experimentation demonstrates the EFSM provides analysts and decision makers a tool for exploring artillery effects in a littoral environment.

14. SUBJECT TERMS
Discrete Event Simulation, Simkit, Anti-Access Area Denial, Artillery, Experimental Design

15. NUMBER OF PAGES 85

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2–89)
Prescribed by ANSI Std. 239–18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

DISCRETE EVENT SIMULATION FOR THE ANALYSIS OF ARTILLERY FIRED PROJECTILES FROM SHORE

James Garrick Sheatzley
Captain, United States Marine Corps
B.S., United States Merchant Marine Academy, 2010

Submitted in partial fulfillment of the
requirements for the degree of

**MASTER OF SCIENCE IN
MODELING, VIRTUAL ENVIRONMENTS, AND SIMULATION**

from the

**NAVAL POSTGRADUATE SCHOOL
June 2017**

Approved by: Arnold H. Buss
Thesis Advisor

Susan M. Sanchez
Second Reader

Peter Denning
Chair, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The purpose of this thesis is to provide decision makers a tool for analyzing the effectiveness of current United States Marine Corps artillery systems conducting anti-access area denial operations. Artillery effects against land targets are documented and understood, but a knowledge gap exists regarding the effectiveness of artillery operations in the littorals. Expeditionary Fire Support Model–Maritime (EFSM) is a discrete event model that simulates current capabilities of Marine Corps artillery systems. Integrating an existing naval convoy model, two proof-of-concept littoral scenarios are presented that represent battalion and regimental artillery task organizations tasked to deny freedom of navigation (area denial) and stop an amphibious naval convoy (anti-access). Results from a designed experiment indicate artillery systems provide commanders a limited area denial capability, and should be employed where naval forces are limited in maneuverability and follow known routes close to shore. Overall, artillery achieve higher destruction rates in the area denial scenario than anti-access scenario. Factors important for successful anti-access area denial operations include unmanned aerial system speed and firing delay of the M777A2 lightweight howitzer. Data produced during experimentation demonstrates the EFSM provides analysts and decision makers a tool for exploring artillery effects in a littoral environment.

THIS PAGE INTENTIONALLY LEFT BLANK

Disclaimer

The reader is cautioned that the computer programs developed in this research may not have been exercised for all cases of interest. While every effort has been made within the time available to ensure that programs are free of computational and logic errors, they cannot be considered validated. Any application of these programs without additional verification is at risk of user.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

List of Figures

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

A2AD Anti-Access Area-Denial

ASM anti-ship missile

CIWS close-in weapon system

COMBATXXI Combined Arms Analysis Tool for the 21st Century

DES discrete event simulation

DoD Department of Defense

ECR effective casualty radius

EFSM-M Expeditionary Fire Support Model–Maritime

EFSS Expeditionary Fire Support System

FEL future event list

FFR fire finder radar

FST fire support team

GUI graphical user interface

HIMARS High Mobility Artillery Rocket System

HVU high value unit

LEGO Listener Event Graph Object

MEU Marine Expeditionary Unit

MOE measure of effectiveness

MOP measure of performance

M777A2	M777A2 Lightweight 155mm Howitzer
NPS	Naval Postgraduate School
P_k	probability of kill
SAM	surface-to-air missile
TLE	target location error
UAS	unmanned aerial system
UML	Unified Modeling Language
USG	United States government
USMC	U.S. Marine Corps
USN	U.S. Navy

Acknowledgments

The author would like to thank Dr. Arnie Buss and Dr. Susan Sanchez for their guidance in the process of developing this thesis. Additionally, the author thanks his wife, for her love, support, and endless patience.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1:
Introduction

The fall of the Soviet Union and the end of the Cold War signaled a change in threats to the United States (U.S.). Blue water navies no longer challenged each other for control of sea lanes, and new threats to open sea lanes emerged as distributed, non-peer, adversaries became able to control strategic choke points around the world. To deter these threats, the United States transitioned its view of sea power by recognizing the need to operate in littoral zones that differ greatly from blue water operations. Littoral zones offer challenges such as limited maneuverability, merchant traffic, and both sea and land-based threats. Joergensen states in [1], "Today littoral warfare is three-dimensional; it has aspects that are more demanding than the blue sea, and it involves environments that favor the defender and the weak." Echoing Joergensen, Kraski in [2] describes how non-peer adversaries such as Somali pirates in the Gulf of Aden, and the Iranian Revolutionary Guard in the Straight of Hormuz, increase the need for maritime security operations in strategic littoral regions. As new threats, such as China, reinterpret law of the sea and deny navigation outside of territorial waters [2], the United States must be ready and able to deter and defend strategic littoral zones around the world.

The history of war at sea tells us that while coastal defense batteries have been effective in denying sea lanes to approaching navies, the number of coastal defense batteries in the world is decreasing. Littoral zones now tend to feature in-depth defenses of surface ships, submarines, aircraft, mines, fast patrol boats, and shore fired anti-ship missiles (ASMs) [1]. The U.S. Navy (USN) and U.S. Marine Corps (USMC) lack many of these systems, and the systems they do have are not designed to operate in a littoral environment. The USN surface and submarine fleets are large and expensive and not well suited for operations in environments of limited depth and maneuverability. The USN does not have a small boat fleet, and mines are of limited use because they deny sea lanes not just to the adversaries, but to friendly and merchant traffic. The United States also lacks a shore-based anti-ship missile (ASM) capability. Additional challenges the USN faces are ship availability and rapid deployment. Aircraft are able to arrive in zone quickly, but are unable to maintain sea lanes indefinitely, and the USN does not have enough ships to project power everywhere at once. If Marines are placed ashore, the USMC is limited to ground-based artillery fires for coastal defense.

1.1 Current USMC Artillery Capabilities

Artillery provides commanders a means to shape battle space, attack high value targets with precision, and maneuver against a suppressed enemy. It plays a critical role in the success or failure of combat operations in a wide range of environments. Though proven in conventional and counterinsurgency operations, artillery systems must continue to provide commanders an adaptable and scalable capability in new environments against emerging threats. As adversary nation states continue to grow their navies and seek to project influence outside their borders, littorals will become contested ground and artillery must be able to conduct operations ranging from denying navigable waterways to repelling an amphibious assault. Current USMC artillery systems have not been tested in this environment, and their capabilities remain unknown. Commanders must possess the knowledge of what capabilities various artillery task organizations provide, and what targeting systems are required to best support artillery units to make informed decisions. With this knowledge, commanders can commit scaled artillery forces to meet the requirements of denying navigable waterways and countering adversary power projection.

Current USMC artillery capabilities consist of conventional artillery, conventional mortars, and rocket artillery. Conventional artillery and mortars are transportable ship-to-shore by medium and heavy lift helicopters, but lack range and require extensive logistical support. Rocket artillery is internally transportable in fixed-wing cargo aircraft that require an expeditionary airfield. Because of the large lift requirement to support distributed artillery operations, rapidly deploying enough units to deny access or control sea lanes is a significant challenge to the USN and USMC. Artillery systems also lack munitions capable of tracking moving targets, and require a system of target locating sensors to be in place to locate and track targets. If the United States desires the capability to control key littoral areas using sea and land-based forces, the surface-to-surface capability gap must be filled. To do this effectively, an understanding of current artillery capabilities is required. This thesis provides a model for studying current and future capabilities of artillery systems in a simulated littoral environment.

1.2 Research Questions

This thesis address the following research questions:

1. What effect does artillery task organization (number and quantity of systems) have on preventing a landing force from reaching its embarkation point?
2. What effect do target locating systems have on preventing a landing force from reaching its embarkation point?
3. Which factors have significant effect on the ability of artillery fired projectiles from shore to destroy different classes of ships?

1.3 Methodology

The Expeditionary Fire Support Model–Maritime (EFSM-M) extends Ali Opcin's SurfaceSim model described in [3] by adding models of current USMC artillery capabilities. SurfaceSim and EFSM-M interoperate to simulate littoral operations. The range of factor values in this thesis use open source data, rather than classified data. Although actual parameter values are not used in the analysis, a designed experiment produces simulation runs of varying artillery task organization, number of rounds fired, target location error, circular error of the impacting rounds, and probability of kill. EFSM-M can be run using actual data. Research methods include the literature review, EFSM-M development, software integration, experimental design, and analysis.

1.4 Scope of Thesis

This thesis is limited to the study of current USMC artillery capabilities, namely: the High Mobility Artillery Rocket System (HIMARS); the M777A2 Lightweight 155mm Howitzer (M777A2); the Expeditionary Fire Support System (EFSS); and USMC targeting systems to include fire finder radar (FFR). Naval vessels are modeled on ships of the USN using open source information. Ship sensors and defensive systems are modeled with an existing sensor model. A designed experiment is used as to demonstrate that the model can provide decision makers with information regarding what system components have the most significant effect on the ability of USMC artillery systems to engage different classes of ships and prevent an amphibious convoy from embarking its landing force.

1.5 Literature Review

A large body of research exists regarding exchanges of gunfire and artillery. Lanchester's area fire model, which explored the effect of exchanging land-based artillery fire at the aggregate level, has been adapted to include stochastic elements and tailored for naval warfare; however, a majority of naval models focus on ship-versus-ship engagements. Historical naval salvo models focus on exchanges of gunfire. More recent models focus on ASM engagements and ship-based anti-air warfare. Fewer models simulate land-sea engagements.

John Schulte in [4] analyzed the effectiveness of ASMs by developing a mathematical model to determine the number of ASMs required to kill one of the following three classes of maritime targets: defenseless targets, defendable targets, and defended targets. Using reports from 25 historical ASM engagements, Schulte determined the probability of hit for seven ASM variants. Schulte concluded that softkill measures, such as decoys and chaff, are extremely effective against ASMs, but hardkill measures are not, and it is disastrous if a ship's defensive system has a leakage rate (probability of missiles passing through defenses) of 0.25 or greater. Not included in his model are large, modern combatant ships such the Arleigh Burke class destroyer, or modern hardkill systems such as the Close-In Weapon System. Schulte notes in [4] that his model, "...can apply to at least three types of U.S. warships, such as the Oliver Hazard Perry frigate (FFG-7 class), a mine countermeasures ship (MCM-1 class), or a new coastal patrol ship (PC-1 class)."

Wayne Hughes offers a methodology for studying the combat characteristics of modern surface warships in [5]. His results emphasize the importance of staying power, the ship's ability to absorb hits and continue fighting, relative to other combat characteristics. To substantiate the results of the salvo model of modern missile combat, Hughes modeled historical naval battles and compared simulation outcomes against historical reports. Ship characteristics studied in the model include: striking power, staying power, counterfire, scouting, soft-kill counteractions, defensive readiness, and training. Hughes's analysis in [5] lead to several conclusions, including: "numerical superiority is the force attribute that is consistantly most advantageous, staying power is least affected by particulars of the battle, weak staying power is the root cause of instability, and instability occurs as combat power grows relative to survivability."

Michael Armstrong recognized that Hughes's deterministic model does not fully reflect the chaotic nature of war. In [6] Armstrong says this about deterministic combat models, "This is a serious concern for a combat model because most warfare is so full of variation and chaos as to be inherently stochastic; the user of a deterministic model therefore risks being misled by its apparent predictability." To improve upon Hughes's salvo model and provide decision makers with better information, Armstrong modifies Hughes's equations to include randomness in the model's parameters. Armstrong replaces the fixed, deterministic variables with independent, identically distributed random variables to account for imperfection in a ship's ability to accurately launch missiles, variability in the number of missiles a ship is able to defeat, and the damage missiles cause. The result of Hughes's deterministic model and Armstrong's stochastic model show similar average surviving force strengths. However, capturing more of the chaos of war, Armstrong's model may better inform tactical decision making. Armstrong, based on analysis of his model, recommends in [6] that, "a force superior in total firepower should try to balance its offensive and defensive capabilities, and then seek battle under conditions of greatest certainty. Inferior forces should seek battle under uncertain conditions and focus on either the offense or defense."

Mahon in [7] enhances Hughes's salvo model to include not only Lanchester's aimed fire model, as Hughes did, but Lanchester's area fire model, producing a model to support the development of naval tactics in a littoral environment. He developed and analyzed two scenarios in [7], looking for "specifically beneficial parameters and trends, which could assist a naval force against Anti-Access Area-Denial (A2AD) threats from shore." Adding complexity to Hughes's model, Mahon included land forces and both direct and indirect fire missiles and shells. One major contribution to the salvo model is the addition of the fractional exchange ratio as a measure of effectiveness. The fractional exchange ratio gives analysts a quantitative measure of force-on-force effectiveness. His findings show that modern naval forces have the ability to attack land based forces, but only if they strike first and have superior scouting capabilities.

Hughes's, Armstrong's, and Mahon's models are aggregate models focusing on fleets of naval forces. Missing from the research is an entity-level stochastic model to be used as a tool to analyze current and future artillery systems in a littoral environment. EFSM-M improves upon previous salvo combat models and seeks to determine what parameters are important for land-based artillery operating as an A2AD weapon system.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 2:
Background

The EFSM-M is a discrete event simulation (DES) model that simulates artillery versus ship engagements in a littoral environment. Rather than develop a model from scratch, the EFSM-M extends an existing naval convoy model, SurfaceSim, and Simkit, a software library for developing DES models, and adds artillery units, projectiles, and sensors to produce a littoral combat model for the analysis of shore fired artillery. Section 2.1 describes discrete event simulation using Simkit for implementing movement and sensing. Simkit's CircularImpactMunition, which approximates artillery fired projectiles, is then discussed, and Chapter 2 closes with a brief description of the SurfaceSim naval convoy model.

2.1 Discrete Event Simulation with Simkit

Simkit is a library of classes and interfaces, written in Java, that support ease of implementation for DES models. Developed by Arnold Buss, Simkit provides a means to code and simulate conceptual models built on Schruben's event graph methodology [8] that model a system by representing the relationship between events being processed and events being scheduled [9]. Discrete event methodology was selected for the EFSM-M because, as Buss describes, the discrete event simulation (DES) worldview supports collecting statistics and the purpose of the EFSM-M model is to determine what artillery systems and sensor characteristics are most critical for successful A2AD operations through analysis. Simkit allows simulation modelers to break complex systems into components through a framework of Listener Event Graph Objects (LEGOs), described in [10], where small component systems are linked together by listeners. The EFSM-M leverages LEGO by breaking fire support and artillery tasks into simple components and connecting them into a complex simulation. This, in turn, makes the EFSM-M extensible, tailorable, and reusable for use in future simulations.

DES models schedule events by placing them on the event list, called the future event list (FEL) in this paper, and executing them at the appropriate time. Time in DES advances only when an event is removed from the FEL and the associated state transitions are executed. It is the process of events being removed and scheduled on the FEL that advances simulation time. Thus, simulated time advances in typically irregular intervals determined by when events are scheduled. The Next Event Algorithm shown in Figure 2.1 diagrams how time advances in DES. As Buss describes in [9], a DES component consists of three

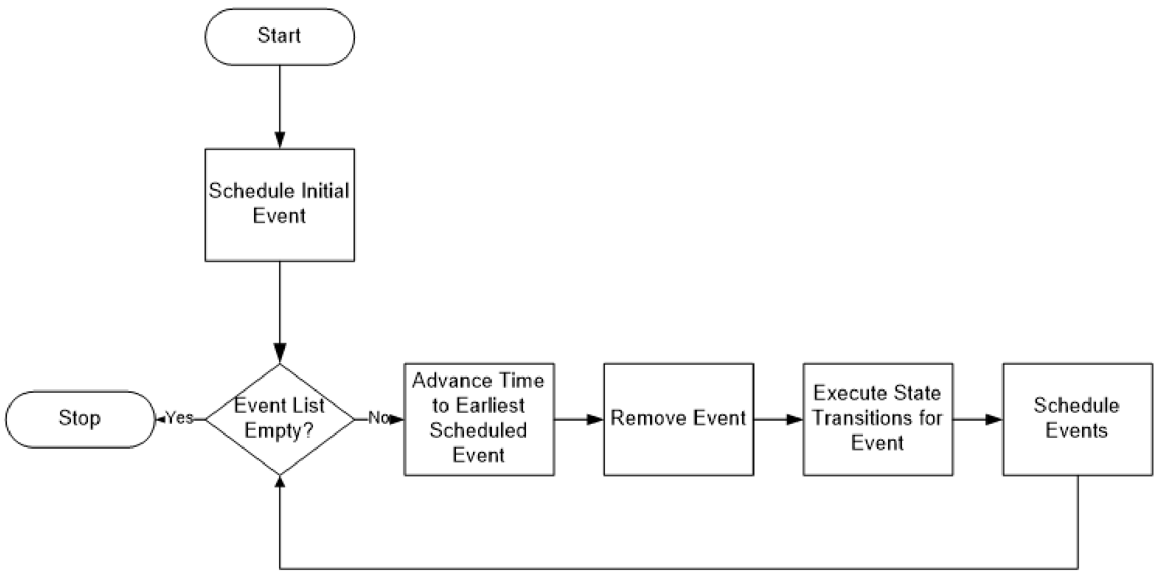


Figure 2.1. Next Event Algorithm. Source: [9]

elements: states, events, and scheduling relationships between events. A state trajectory consists of the successive values of a state as it evolves over time. DES state trajectories are piecewise constants, with all state transitions occurring at events. Simkit allows the modeler to determine the state trajectory by firing a “property change event" at each state transition. This allows property change listener objects, such as those gathering statistics, to compute their respective measures from the state trajectories.

Simkit manages events and reduces user interaction with the FEL with its SimEntity interface, SimEntityBase abstract class, and simkit.Schedule class [11]. The simkit.Schedule class uses Java's SortedSet class to manage events in time sorted order on the FEL. Simulation modelers do not place events directly on the FEL; Simkit's SimEntityBase abstract class performs FEL interactions, and so reduces the possibility of user error affecting simulation execution. By implementing all abstract methods of the SimEntity interface, simulation modelers meet Simkit's requirement for scheduling events.

Defining all possible state transitions and events for a large and complex system with a single event graph can quickly become overly complex. Simkit is built with two listener patterns to link LEGOs: SimEventListener and PropertyChangeListener. As Buss states in [10], this “allows the simulation modeler to create simulation components that encapsulate Event Graph Logic, then connect the components together to create larger models of greater complexity.” The SimEventListener pattern allows one component to “hear” events in another component and execute a corresponding event in the listening component. An example of the SimEventListener pattern is shown in Section 2.1.1. The PropertyChangeEvent supports the DES concept of state transition. Buss describes how each time a state transition occurs, an associated PropertyChangeEvent fires. PropertyChangeEvents are heard by interested components and notify the component that the state of an object has changed. The PropertyChangeListener pattern is key for allowing what Buss [10] describes as the “non-invasive” collection of statistics.

2.1.1 Moving and Sensing

Many simulators believe that DES cannot simulate motion because of its event-based time-advance method and the fact that all DES state trajectories must be piecewise constant. However, Buss and Sanchez in [12] debunk this common misconception and present a methodology for simple linear motion. Their methodology is applied to EFSM-M to simulate the motion of naval forces, projectiles, radar, and their interactions. In a DES worldview, position is not an explicit parameter of an object. The position of an object is calculated when needed. Buss and Sanchez describe this calculation of position as “implicit state." To calculate the position of an object, the object’s starting position, time that movement started, and a velocity vector must be known.

Basic movement and detection is implemented in Simkit using LEGO consisting of four main components:

1. Mover Component
2. Sensor Component
3. Referee Component
4. Sensor-Target Mediator Component

The mover component is responsible for determining position based on an equation of motion. Simulation modelers create moving objects by implementing the Simkit Mover interface. The DES state consists of the initial conditions of the equation of motion. The sensor component has two functions which are to maintain a list of all current contacts and store the variables required for the equation of motion. Figure 2.2 are event graphs for a simple mover and sensor.

The referee component schedules enter and exit range events, and sensor-target mediators schedule detection and undetection events. Together, the referee and sensor-target mediator components allow the simulation modeler to keep "ground truth" information from mover and sensor components by not allowing sensor components to schedule their own enter and exit range events or to implement the equation of motion [12]. Sensor-target mediators are created for each sensor target pair, allowing simulation modelers to implement any number of detection algorithms without effecting individual sensors or movers.

Figure 2.3 shows the LEGO listener pattern and event graphs for the mover component and referee component. The referee component is "listening" to the mover component for StartMove and EndMove events. If the referee component determines that the mover will enter the range of a sensor registered with the referee, the referee will schedule the enter and exit range events for the sensor. The mediator component, shown in Figure 2.4, schedules detection and undetection events by listening to the referee for enter and exit range events. Based on the equation of motion and type of sensor component (cookie cutter sensor, constant time sensor, constant rate sensor), the mediator schedules detection and undetection events for the sensor.

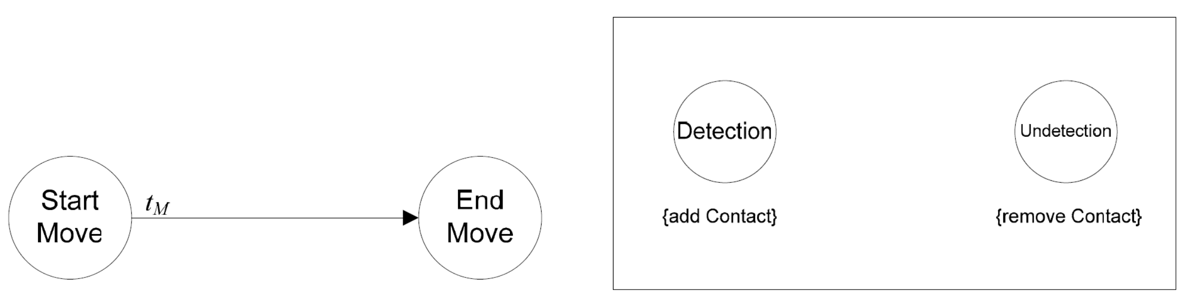


Figure 2.2. Mover and Sensor Component Event Graphs. Source: [12].

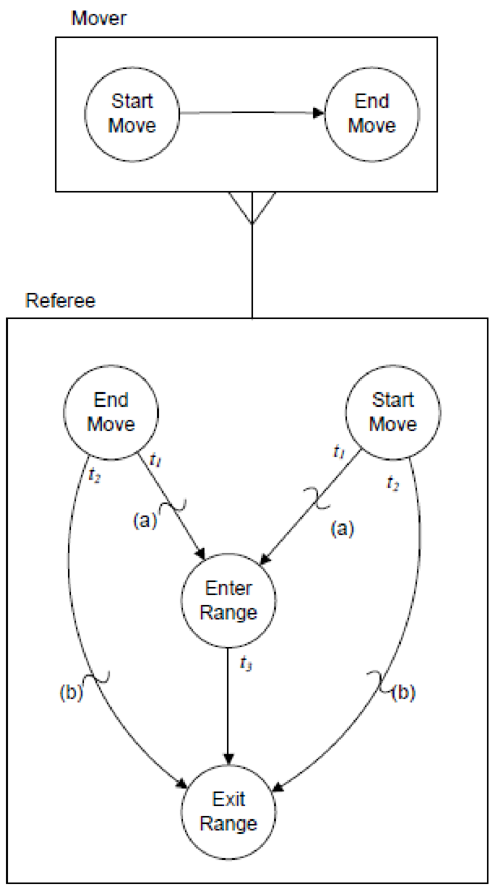


Figure 2.3. Mover Referee SimEventListener Pattern. Source: [12].

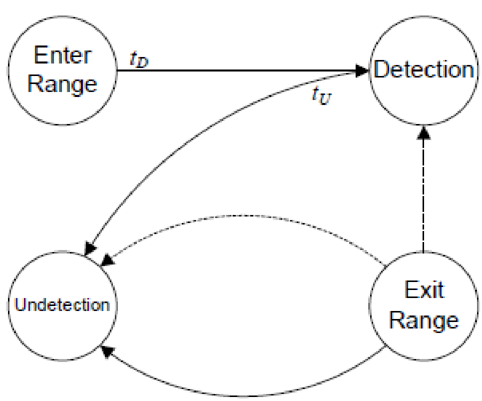


Figure 2.4. Mediator Component Event Graph. Source: [12].

2.1.2 Circular Impact Munition

Simulations such as the USMC Combined Arms Analysis Tool for the 21st Century (COMBATXXI) use Simkit libraries to provide functionality and, as a result, additional class libraries have been added to Simkit. EFSM-M uses Simkit's CircularImpactMunition to approximate artillery fired projectiles and rockets because artillery munitions have a circular impact radius. Implementing a CircularImpactMunition approximates the interaction of a munition defined by a circular blast radius with a target. It also handles movement of the munition from the munition's firing point to impact location. Upon impact, a defined blast radius is checked against all targets registered with the munition to determine if a target is located within the blast radius. A munition target adjudicator, created by the simulation modeler, determines the damage to the target. The advantage of using CircularImpactMunition for EFSM-M is the ability to quickly create new munitions without altering existing source code.

2.2 SurfaceSim

The EFSM-M is an extension of Ali Opcin's SurfaceSim model which simulates the anti-air warfare systems of combatant ships conducting naval convoy operations. Further information regarding Opcin's SurfaceSim model is available in [3]. SurfaceSim is a stochastic model of the anti-air warfare weapon and sensor technology of Turkish frigates. Opcin developed three convoy scenarios and conducted a designed experiment to determine which factors have the greatest effect on the primary measure of effectiveness (MOE): the survival of a high value unit. The EFSM-M extends SurfaceSim by adding components that model land-based artillery systems, land-based sensors, and aerial sensors. Combined, the SurfaceSim model and EFSM-M simulate engagements between land-based artillery and naval convoys in a littoral environment.

Implementing EFSM-M required some minor modifications to SurfaceSim to enhance its flexibility. Entity types required for target recognition and target-munition adjudication were added to SurfaceSim's Type Enumeration class and Adjudicator class. In addition, target-munition interactions in SurfaceSim and EFSM-M are adjudicated differently. The SurfaceSim model uses a simulation modeler-defined Adjudication class to determine munition-target interactions, where as EFSM-M uses Simkit's CircularImpactMunition class and Target interface. Vessel class, a subclass of SurfaceSim's Ship class, is required to implement the Target interface in EFSM-M.

EFSM-M and SurfaceSim take full advantage of Simkit to implement DES. Interactions between sensors and movers, as well as interactions between targets and munitions, is handled by Simkit classes. This allows model developers to focus on the events, state transitions, and relationships required to build models such as EFSM-M and SurfaceSim. The LEGO and listener patterns, described in Section 2.1, support collecting statistics and allow EFSM-M and SurfaceSim to combine into a single littoral combat model. Chapter 3 explains how Simkit, SurfaceSim, and their components come together to form the artillery model, EFSM-M.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 3: Expeditionary Fire Support Model–Maritime

The EFSM-M simulates artillery engagements beginning with a target being sensed by an observer, radar, or unmanned aerial system (UAS) and ending with the adjudication of a munition impact. The simulation approximates how artillery missions are processed in the Marine Air Ground Task Force and includes USMC units such as an infantry fire support team, fire support coordination center, artillery battery, target acquisition platoon, and individual artillery systems. Chapter 3 begins with a description of the mission processing components of EFSM-M and closes with detailed class descriptions. Event graphs and model descriptions are only provided for components native to EFSM-M. Information on Simkit classes can be found in the Simkit Documentation.

3.1 Mission Processing

Mission processing begins and ends with the FireSupportCoordinator. Figure 3.1 shows how the FireSupportCoordinator processes missions. The first event classifies the type of Contact detected in the Classify event and schedules a Counter Fire or Engage event depending on the type of Contact. The FireSupportCoordinator schedules a Counter Fire event, for incoming GunRounds, or an Engage event if the Contact is a Vessel. Counter Fire protects friendly forces and is scheduled with a higher priority than Engage events. Both Counter Fire and Engage events schedule Send Mission events for a selected ArtilleryUnit. Figure 3.1 shows how Counter Fire and Engage events directly schedule Cease Load and Send Mission events in the ArtilleryUnit class. Feedback to the FireSupportCoordinator from the ArtilleryUnit is achieved through the SimEventListener pattern shown in Figure 3.2. Targets are re-engaged if they are alive and still visible to one of the sensors registered to the FireSupportCoordinator.

ArtilleryUnit instances are responsible for processing two events during mission processing: Cease Load and Send Mission. Because Counter Fire missions are processed immediately by the first available and in-range ArtilleryUnit, the Cease Load event cancels all current fire missions being processed by directly scheduling StopShooting events and cancelling future Shoot events. The second event processed by ArtilleryUnit instances, as shown in

Figure 3.3, is the Send Mission event which determines how to engage a specified Contact type before directly scheduling Shoot events for each Artillery instance belonging to the ArtilleryUnit.

Artillery instances schedule events Shoot, StopShooting, and begin the mission processing feedback loop by scheduling the Rounds Complete event. Figure 3.4 shows how Artillery instances directly schedule Fire events for ArtilleryMunition. ArtilleryMunition is a subclass of CircularImpactMunition explained in 2.1.2. Shoot events continue to schedule until the number of volleys specified by the ArtilleryUnit are fired, then a StopShooting event is scheduled. The SimEventListener pattern in Figure 3.2 shows how Artillery instances initiate the mission complete feedback loop.

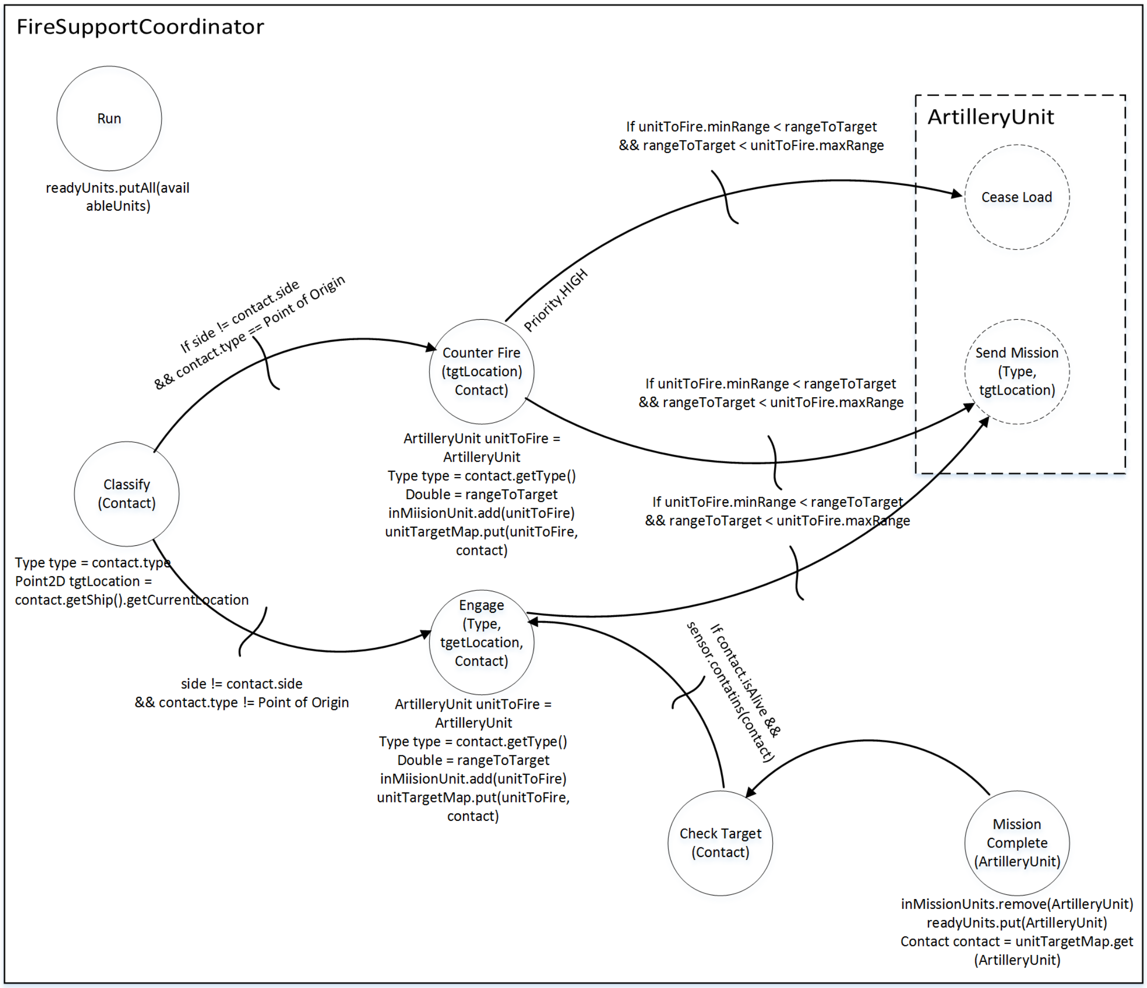


Figure 3.1. FireSupportCoordinator Class Event Graph Diagram

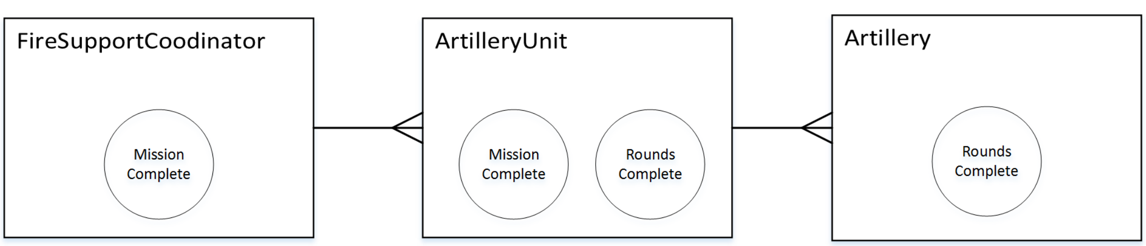


Figure 3.2. Mission Processing SimEventListener Pattern

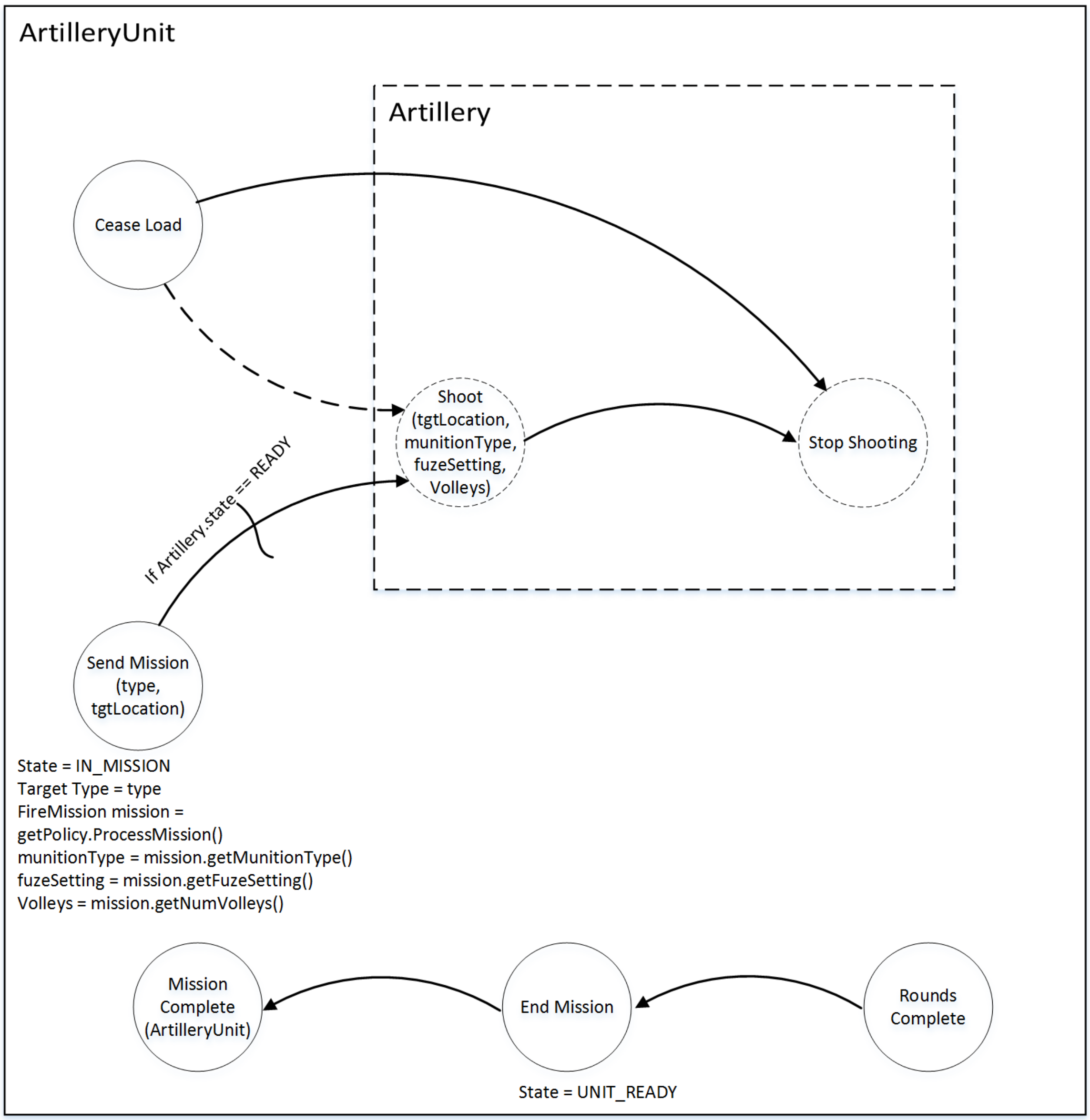


Figure 3.3. ArtilleryUnit Class Event Graph Diagram

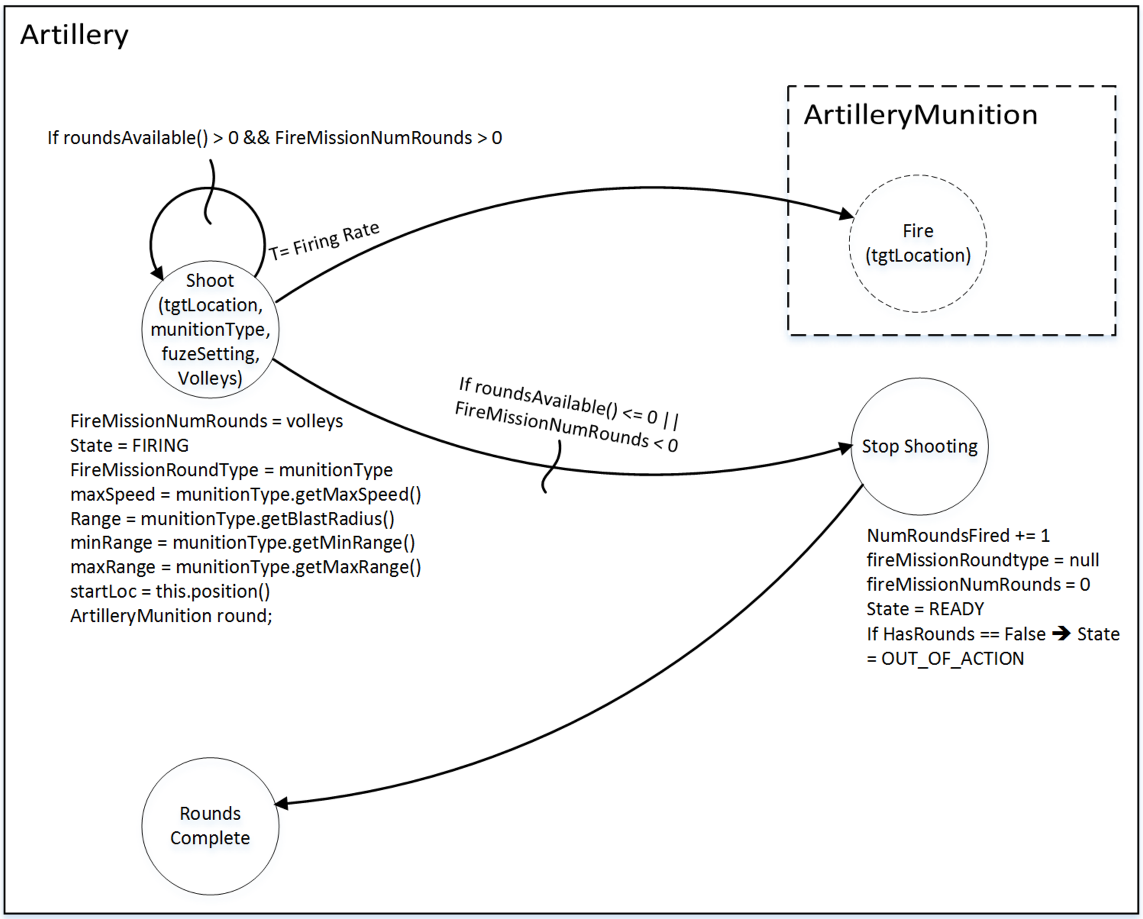


Figure 3.4. Artillery Class Event Graph Diagram

3.2 Sensors

The EFSM-M uses two types of sensors, CookieCutter and ConstantTime, to approximate human observers, UASs, and fire finder radar. CookieCutter sensors are simple and detect movers immediately upon entering the sensor's range. ConstantTime sensors detect movers after a specified time delay. To incorporate target location error (TLE) into sensors, the EFSM-M uses SurfaceSim's CookieCutterSensor and ConstantTimeSensor classes which modify Simkit's implementation of CookieCutter and ConstantTime sensors. Error is induced by by adding a Rotated Bivariate Normal random vector to a target's true location. Inducing randomness into sensors is important because it better approximates human observers, UASs, fire finder radar, and enables analysis of the target location error's affect on MOEs.

3.3 Class Descriptions

Section 3.3 discusses the purpose and function of classes in EFSM-M. Unified Modeling Language (UML) diagrams for classes unique to EFSM-M and their superclasses show the variables and methods available to simulation modelers. Not all model functionality is used in the simulation. Class-specific functionality that is not used in the simulation is provided in each class description.

3.3.1 FireSupportCoordinator

The FireSupportCoordinator is the mission processing decision maker. It hears detection events from sensors and determines if the target should be engaged by an artillery unit. The FireSupportCoordinator then determines what artillery units are available and in range of the target. Mission data is sent by the FireSupportCoordinator to available and in-range units to conduct the mission. Instances of FireSupportCoordinator are not visible in the simulation and cannot be targeted by enemy units. The sole purpose of the FireSupportCoordinator is to receive sensor data and determine which ArtilleryUnits should engage a target. Figure 3.5 shows variables and methods for FireSupportCoordinator.

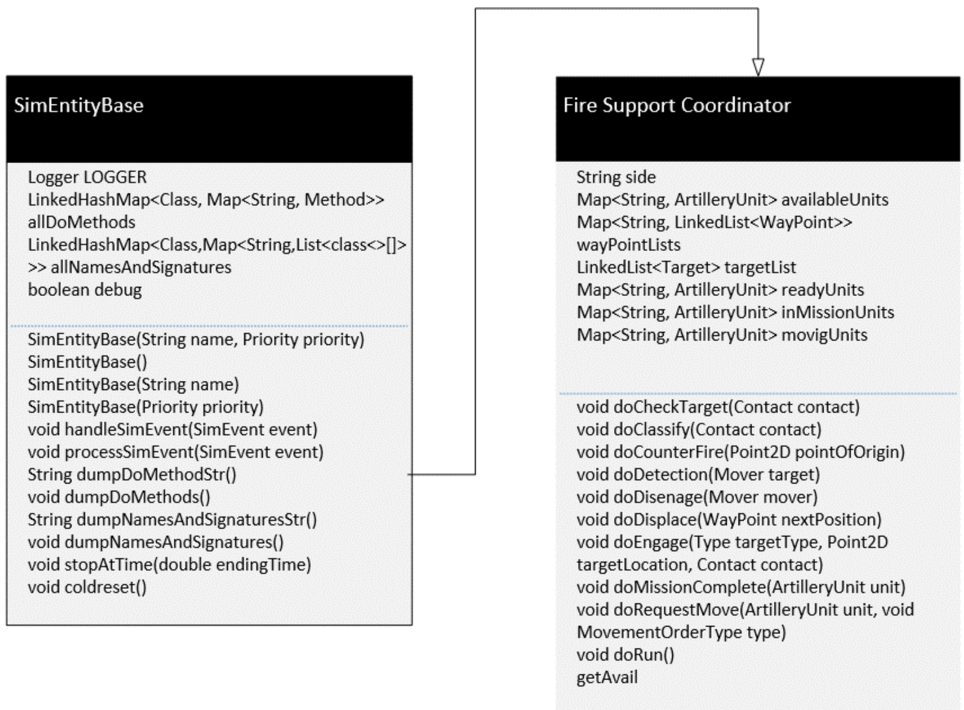


Figure 3.5. FireSupportCoordinator UML Class Diagram

3.3.2 ArtilleryUnit

The ArtilleryUnit class the largest moveable, land based, entity in the EFSM-M. Its purpose is to receive missions from the FireSupportCoordinator, determine the number and type of round to fire for a target type, and order the individual artillery pieces in a firing battery to fire. ArtilleryUnits have the ability to move on order from the FireSupportCoordinator or schedule their own move event if displacement conditions are met. Displacement conditions are: number of missions fired from a single location, and when the unit's efficiency parameter value drops below a threshold value. Each ArtilleryUnit processes one mission at a time, and when complete, report mission complete and their current state to the FireSupportCoordinator. Figure 3.6 shows variables and methods for ArtilleryUnit.

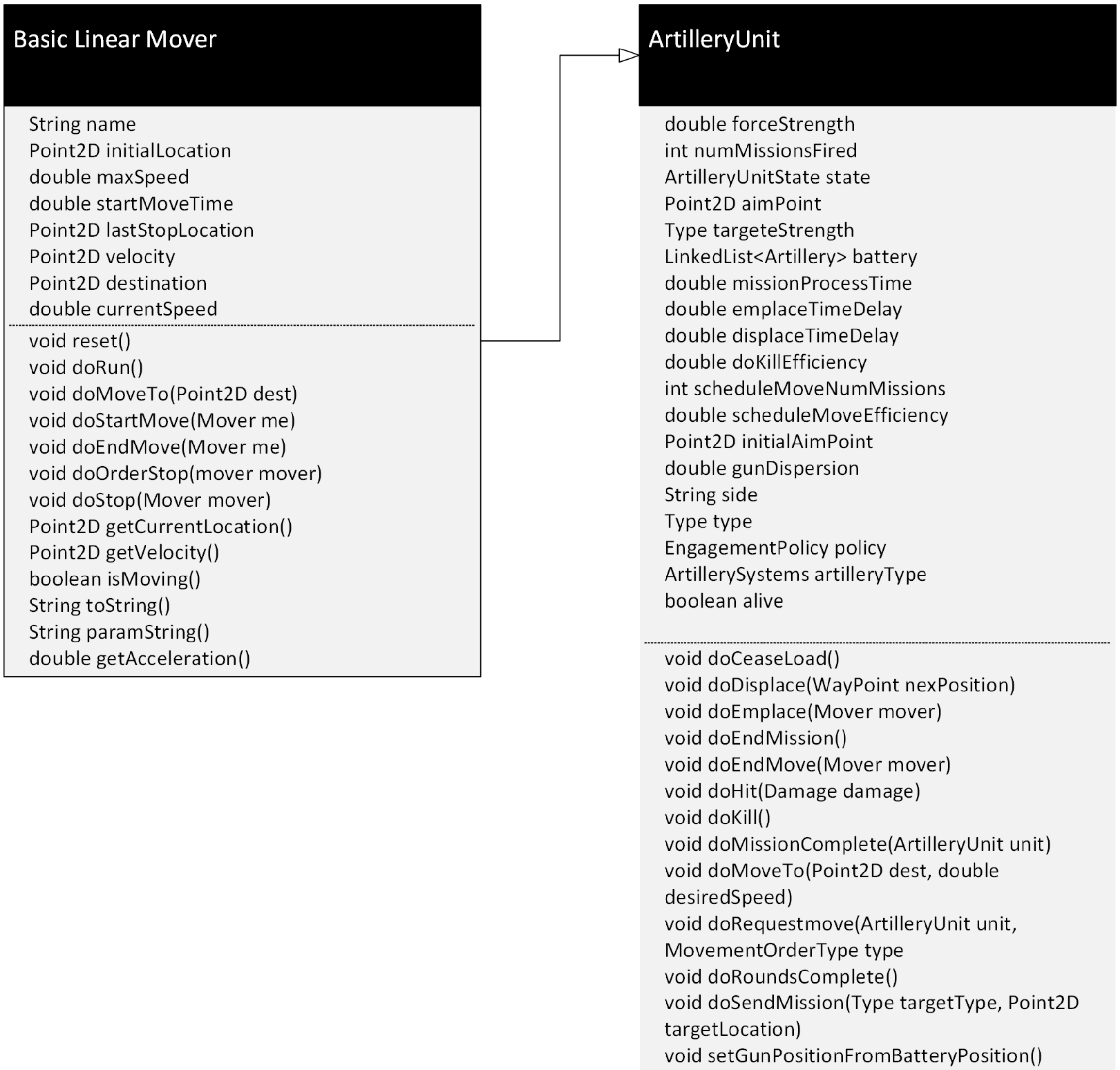


Figure 3.6. ArtilleryUnit UML Class Diagram

3.3.3 Artillery

The Artillery class is responsible for executing fire missions as directed by an ArtilleryUnit. Artillery instances have the ability to schedule Shoot, Stop Shooting, and Kill events. Although, Artillery instances are capable of moving independently, this functionality is not currently used in EFSM-M; the location of artillery instances is calculated in relation to its ArtilleryUnit. The number and type of Artillery instances created is left to the simulation modeler. Current Artillery system types available to the simulation modeler are: M777A2, EFSS, and HIMARS. Figure 3.7 shows variables and methods for the Artillery class.

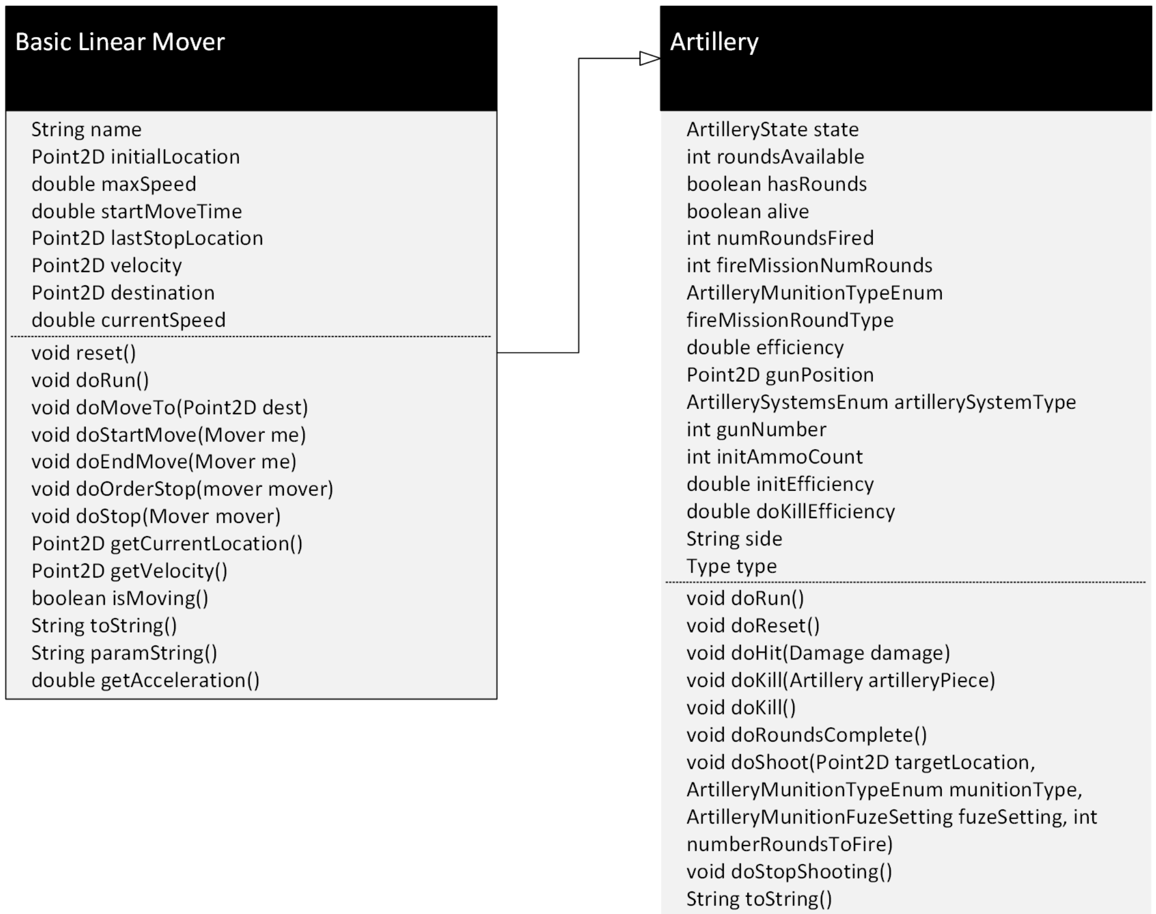


Figure 3.7. Artillery UML Class Diagram

3.3.4 Artillery Munition

ArtilleryMunitions are dynamically created simulation entities that model the characteristics of artillery projectiles, namely movement and target-munition interactions. User defined ArtilleryMunition instances can be created to approximate the range, velocity, and blast characteristics of any munition that produces a circular blast pattern upon detonation. It is important to note that minimum and maximum ranges of artillery systems are set by the minimum and maximum range parameters of the ArtilleryMunition, and is thus independent of the artillery system. Figure 3.8 shows variables and methods for ArtilleryMunitions.

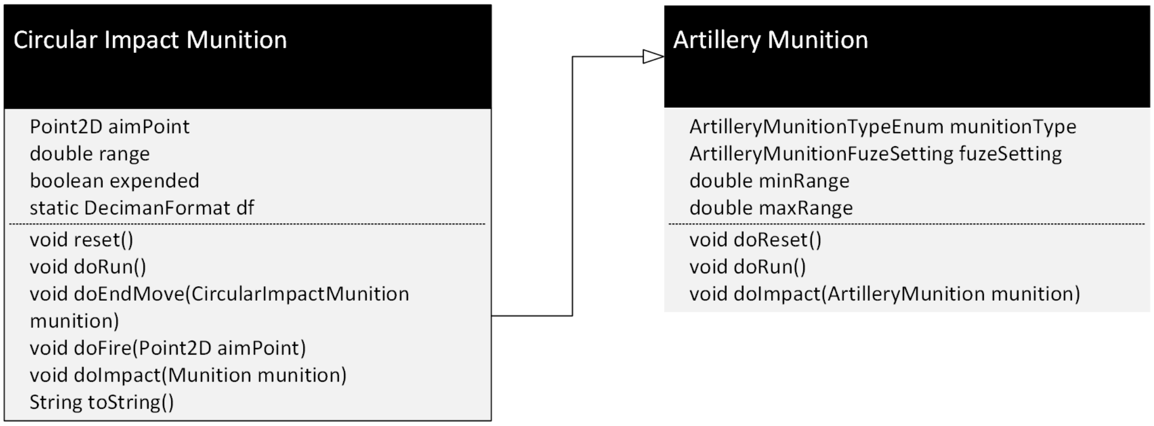


Figure 3.8. Artillery Munition UML Class Diagram

3.3.5 Fire Support Team

The FireSupportTeam models the capabilities of a Marine Infantry Company's fire support team (FST), comprised of observers with hand-held optics for locating and observing targets. FireSupportTeam implements the Simkit Target interface, and may be engaged by enemy units. FireSupportTeam reports contacts to the FireSupportCoordinator.

3.3.6 Fire Finder Radar

The FireFinderRadar class detects incoming artillery and ship's gun rounds based on two criteria: a simulation modeler-defined constant time delay, and a probability of correctly identifying the rounds point of origin. Although FireFinderRadar can be targeted by enemy units, ships do not engage FireFinderRadar in this experiment. Contacts are reported to the FireSupportCoordinator.

3.3.7 Unmanned Aerial System

UnmannedAerialSystems actively search an area by patrolling between, simulation modeler defined, waypoints. Detection events are reported to the FireSupportCoordinator. The UnmannedAerialSystem class implements Simkit's Target interface, but does not currently support ships engaging UnmannedAerialSystem objects with surface-to-air missiles (SAMs).

3.3.8 Engagement Policy

The EngagementPolicy class allows to the simulation modeler to define the engagement criteria ArtilleryUnits apply to engaging targets. Currently, EngagementPolicy supports defining policies for M777A2, HIMARS, and EFSS against different types of targets. The result of calling EngagementPolicy's ProcessMission method is the creation of a FireMission object which contains the type of munition, type of fuze, and number of volleys to fire at a target.

3.3.9 Vessel and Vessel Gun Round

Vessel and VesselGunRound implement Simkit's Target interface and subclass SurfaceSim's Ship and GunRound classes, respectively. Subclassing Ship and GunRound allows for the Target interface to be implemented without changing SurfaceSim's source code. See [3] for a detailed explanation of SurfaceSim's Ship and GunRound classes.

3.3.10 Munition-Target Adjudicators

Munition-Target adjudicators are a subclass of Simkit's Adjudicator class and responsible for adjudicating interactions between various types of munitions and targets. Interactions are simulation modeler defined. The EFSM-M currently has one Munition-Target adjudicator class called High Explosive-Ship Adjudicator. Its purpose is to determine the outcome of interactions between high explosive projectiles and ships. Additional Munition-Target adjudicators may be added to the simulation as required.

3.3.11 Enumeration and Enum Base Classes

A disadvantage to using Java Enum Types is the inability to change the values of Enum Type parameters while conducting a designed experiment. However, a desirable characteristic of enum types is the ability to enable a variable to be a set of predefined constants [13]. Enum types prevent the user from creating object types not recognized by the model. Simkit's EnumBase class is similar to a Java Enum type, but supports changing EnumBase parameters during the execution of a simulation run. EFSM-M uses two EnumBase classes, the ArtilleryMunitionTypeEnum and ArtillerySystemsEnum to define ArtilleryMunition and Artillery objects. These two classes ensure that only artillery systems and munitions recognized by the model are instantiated at run-time.

3.4 Model Assumptions

EFSM-M attempts to capture the important characteristics of artillery systems and target locating sensors, but the entities in the model are only approximations of real systems. As such, assumptions are made to account for the characteristics not represented in EFSM-M. Assumptions and complete factor descriptions for Ali Opcin's SurfaceSim model are described in [3]. A complete list of EFSM-M factor names and descriptions appear in the Appendix. EFSM-M assumptions are:

1. Simulation initial conditions described in A.1 and A.2 are the model's primary assumptions. Initial conditions are based on reasonable values from unclassified, open source data.
2. Kill events are scheduled if a random draw from a uniform distribution is less than the user defined probability of kill (P_k) value.
3. Damage calculations follow a uniform distribution truncated between minimum and maximum simulation modeler-defined damage values.
4. The fire support coordinator cannot be targeted.
5. Artillery units and systems are stationary. Unit locations are set by the simulation modeler.
6. The number of artillery units is static for each simulation run.
7. Initial efficiency for artillery units and systems is 100 percent.
8. Artillery units cannot be targeted. Unit efficiency is a function of artillery system efficiency.
9. Artillery units process one mission at a time in order of target detection.
10. Artillery system range is determined by the artillery munition.
11. Artillery munition ballistics are not calculated. Time to impact is calculated using distance to target and munition speed.
12. Artillery munitions cannot be destroyed by a vessel's anti-air warfare systems.
13. Artillery munitions are aimed at a single point; rounds are not distributed according to a sheaf.
14. All artillery munitions have a circular impact radius.
15. Height of burst for artillery munitions is not considered in the model.
16. Artillery munition fuze settings are not included in the experimental design.
17. Fire support teams and UASs will correctly identify 100 percent of targets detected.

18. A fire finder radar detects rounds if probability of detection is greater than a random draw from a uniform distribution.
19. Fire finder radar and fire support teams are static.
20. Fire finder radar cannot be targeted by vessels.
21. Targets are engaged immediately when detected by a sensor and in range of an artillery unit.

EFSM-M models current USMC artillery operations from sensor to shooter. Additional capabilities can be added at any time. EFSM-M reads initial conditions from a file, making it easy for the modeler or analyst to simulate different variants without modifying the code.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 4: Proof-of-Concept EFSM-M Experiment

The purpose of EFSM-M is to provide decision makers a tool for analyzing current Marine Corps artillery systems performing anti-access and area denial operations, and for exploring future system design alternatives. Building a model that supports experimental design is advantageous for analyzing artillery systems and other complex systems because an appropriate design can examine a large number of factors effecting operational performance. Vieira et al. explain in [14] that new experimental designs, such as the nearly orthogonal Latin hypercubes explained in [15] and nearly orthogonal-and-balanced, mixed designs [14] permit model exploration that is preferred to optimization methods that seek a single, "best alternative" [14]. In a complex littoral environment with advanced artillery and ship systems it is difficult to imagine one factor that determines mission success. Experimental design takes a holistic view to determine how multiple factors and their interactions contribute to the outcome of a simulation.

4.1 Simulation Scenarios

As a proof-of-concept for EFSM-M, two scenarios simulating area denial and anti-access operations are presented. Each scenario features a six vessel naval convoy threat with unvarying weapon characteristics. The speed is the only characteristic of the naval convoy that changes across simulation runs. The number of sensors in each scenario is the same, but the characteristics of those sensors change. Sensors available to the defender include three FSTs, two UAS, and one FFR. Factors in Scenario One and Two are the same.

4.1.1 Scenario One

Scenario One is a restricted navigation scenario simulating area denial operations. The naval convoy is limited in speed and maneuverability while transiting through a choke point. Defending forces are established in positions to engage the convoy at a turn, enabling all artillery systems to mass fires on the transiting convoy in a single location. The advantage lies with the defender in Scenario One, because the defender knows the convoy's route but the positions of the defender (land forces) are unknown to the convoy. Sensors are located along known routes to best observe the engagement area. The area denial experiment is repeated twice: once with an artillery battalion (three M777A2 batteries, one HIMARS battery, and one EFSS battery) and once with a regiment of artillery (nine M777A2 batteries, one HIMARS battery, and one EFSS battery). Figure 4.1 and Figure 4.2 show the artillery formations and naval convoy route for Scenario One.

4.1.2 Scenario Two

Scenario Two is a simulation of anti-access operations. The goal of the attacking naval convoy is to disembark its landing force before the high value unit (HVU) is destroyed. The attacker is approaching from the open ocean to a position close enough to shore for landing craft to reach shore and return to the convoy. The defender in this scenario is forced to cover a large area of land and the attacker is able to maneuver in and out of the engagement area. The naval convoy will make a series of maneuvers parallel to shore at varying distances from shore. Landing craft are not simulated. Scenario Two is run twice with artillery battalion and regimental task organizations. Figure 4.3 and Figure 4.4 show artillery formations and naval convoy route for Scenario Two.

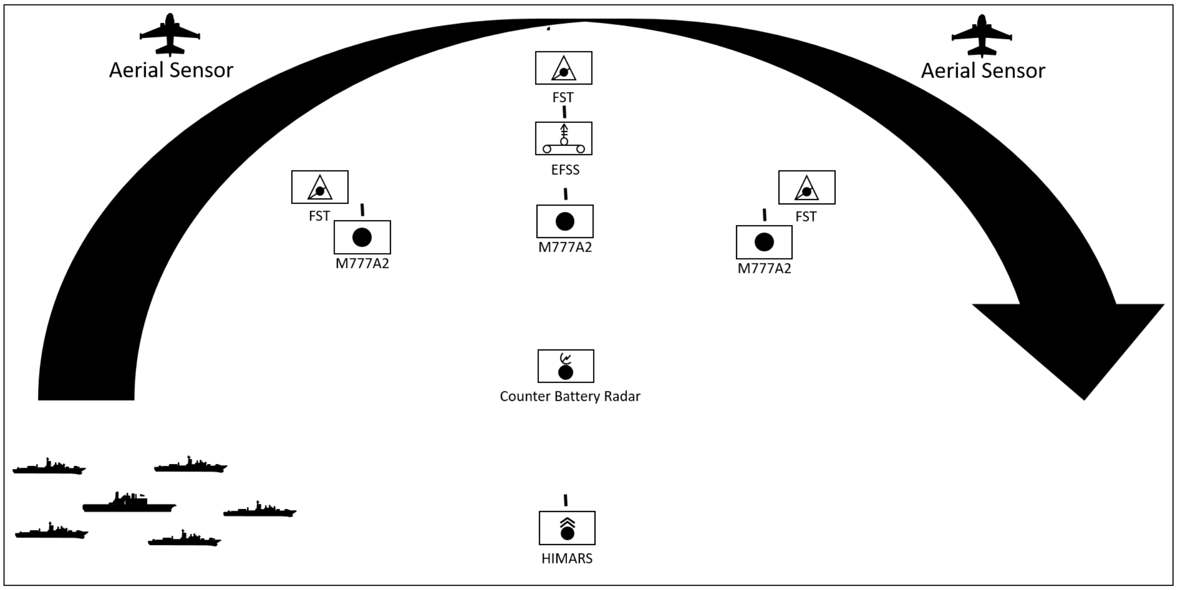


Figure 4.1. Scenario One Artillery Battalion Task Organization Formation.

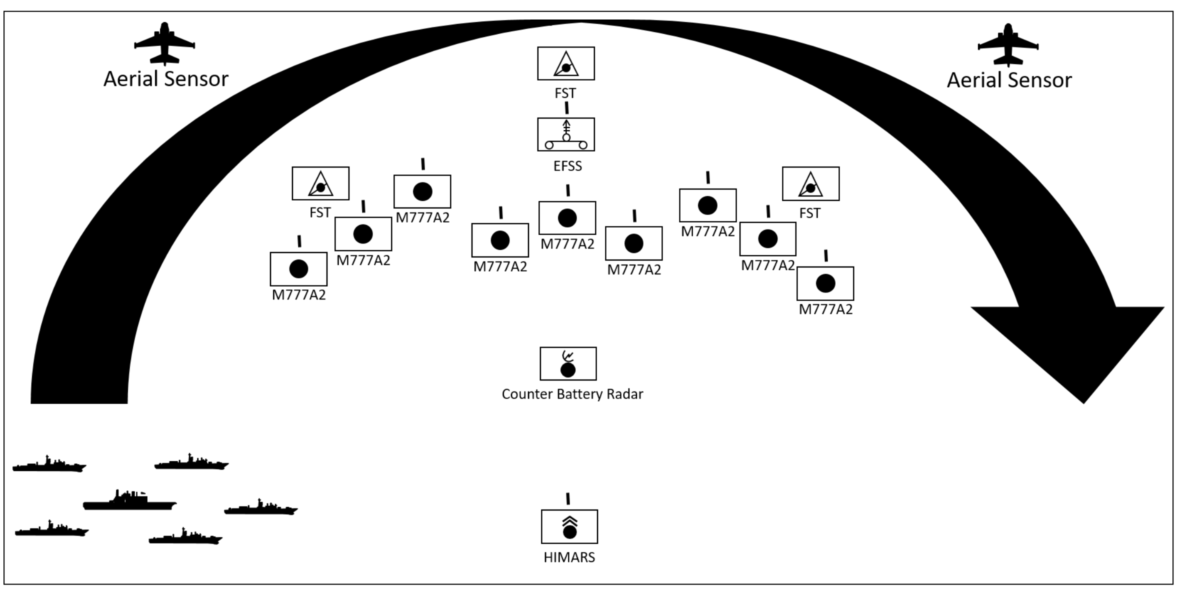


Figure 4.2. Scenario One Artillery Regiment Task Organization Formation.

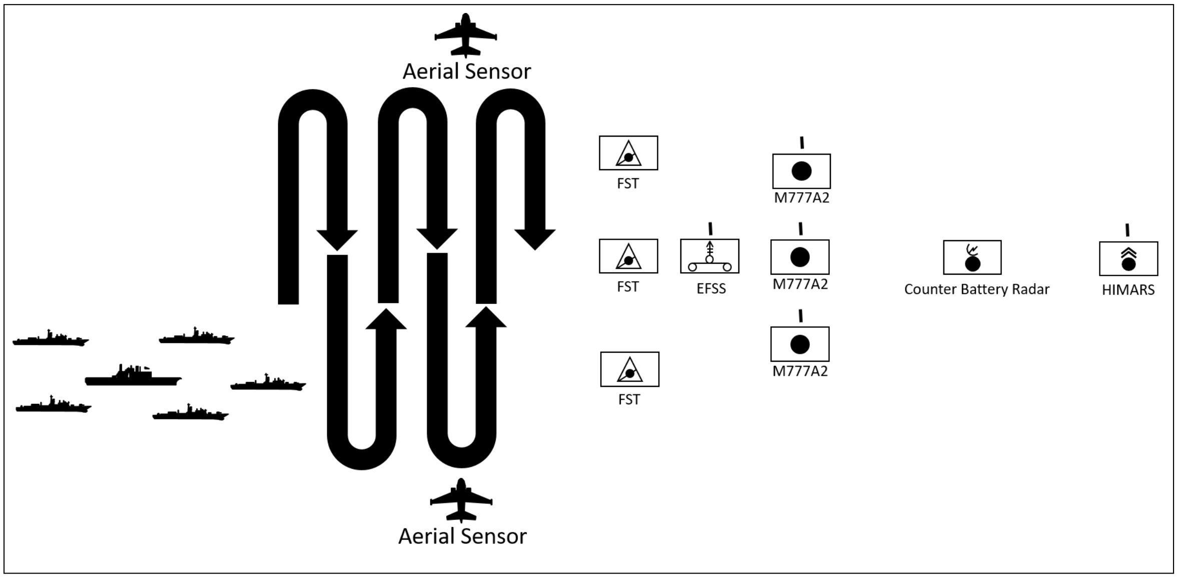


Figure 4.3. Scenario Two Artillery Battalion Task Organization Formation.

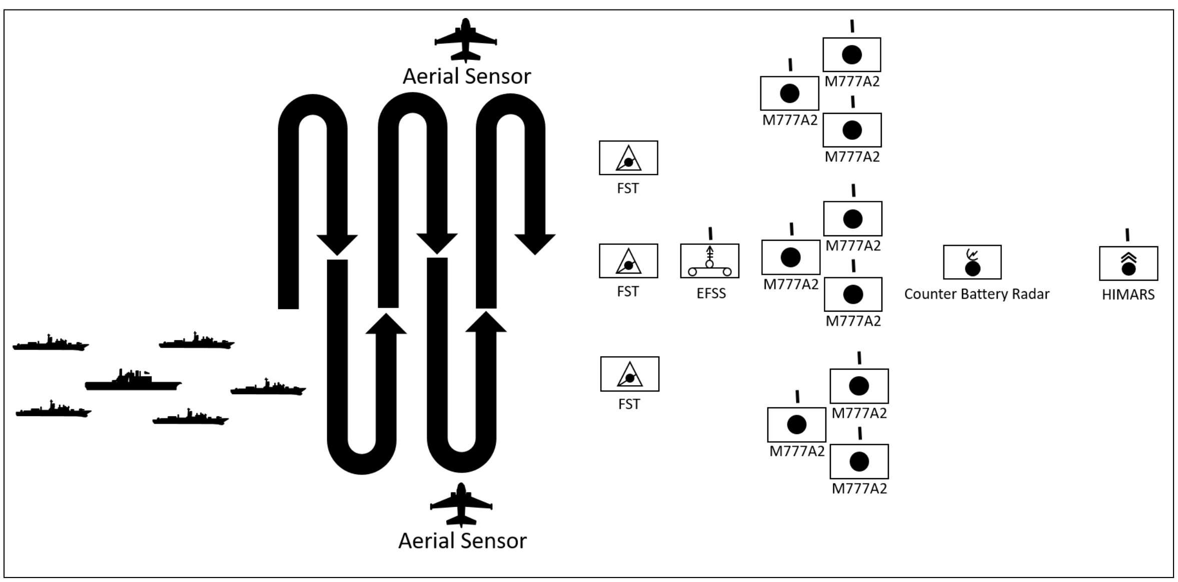


Figure 4.4. Scenario Two Artillery Regiment Task Organization Formation.

4.2 Measures of Effectiveness

The goal of experimentation is to determine which factors have the greatest effect on the MOEs. For both scenarios, the primary MOE is the survival of the HVU. The result is binary for each simulation run. The HVU in each scenario is a vessel equivalent to a USN LHD amphibious assault ship. HVU destruction occurs if its efficiency (remaining strength) parameter falls below a threshold value or a randomly drawn variable is less than a munitions P_k value for the HVU ship class. Simulation modelers define HVU efficiency threshold and desired distribution for random number generation. Ships within the blast radius of a munition, but not killed, are assessed damage based on a user-defined damage function. The damage function reduces a vessel's efficiency by a random number of percentage points truncated between the minimum and maximum damage values for a munition. In this experiment the resulting output value of the damage function is a power of ten less than the input value. Simulation modelers can change the damage function by creating new Munition-Target adjudicators.

4.3 Experimental Design Selection

EFSM-M is a complex model featuring continuous and discrete factors. Its systems include both surface-fired and ship-fired projectiles and missiles, counter-battery sensors, and multiple target acquisition systems. An appropriate experimental design must accommodate all data types and be appropriate for a simulation with a large number of factors. Though not all factors of SurfaceSim and EFSM-M are included in the experimental design, more than 150 factors may be included in future experiments. An efficient, nearly orthogonal-and-balanced, mixed design is used because, according to Vieira et al. in [14], the design supports the following model requirements:

- It can handle categorical, discrete, and continuous factors with multiple levels.
- It has low pairwise correlation (less than 0.05) between any two factors.
- It may be analyzed using a variety of parametric and non-parametric statistical and graphical methods.

Low pairwise correlation is important because it allows factor levels to be analyzed independently [14]. Parametric and non-parametric statistical methods simplify the presentation of results and are commonly used and accepted. The design selected for the 50 factors explored in this thesis is a nearly orthogonal-and-balanced, mixed design with 512 design points. EFSM-M allows analysts to substitute experimental designs without making software changes as long as there are no additions to the list of factors: the only requirement is to update the name and file location of the new experimental design in the model's main method. Adding factors and MOEs will require minor changes to the model. The file scanner, which reads the experimental design into EFSM-M, will require updating to read in additional values from the design worksheet and set model parameters to the appropriate design point.

Factors in this proof-of-concept EFSM-M experiment include: characteristics of artillery munitions, artillery systems, sensors, UASs, threat vessel speed, number of artillery units, task organization of artillery units, and number of artillery volleys to fire at a target. A complete list of factor names, descriptions, and associated ranges of values explored in this thesis appears in the Appendix. In all, there are 50 quantitative factors (including two discrete-valued factors). Scenarios One and Two are run twice with 1000 iterations per design point for a total of 2,048,000 simulation runs.

4.4 Equipment

EFSM-M simulation used the following computer software and hardware:

- Microsoft Windows 10 Operating System
- Intel (R) core (TM) i7-5820k CPU 3.30GHz
- 16.0GB RAM

4.5 Analysis, Results, and Recommendations

Results for each simulation scenario were analyzed using JMP Pro 13 (SAS Institute Inc. 2016) analytical software. Raw data for each simulation run is summarized over replications for each design point, and partition trees on the summary dataset are used to determine which factors are the best predictors of the MOE. Table 4.1 summarizes the mean HVU destruction rate for each scenario. The comparison of HVU destruction rates between Scenarios One and Two is expected. The advantage in Scenario One is with the defender, and results in higher HVU destruction rates than Scenario Two for both battalion and regimental task organizations. Comparing task organizations for Scenario One produces an interesting result: the HVU is destroyed 11.2% less often, even though the defender has an additional six M777A2 units. HVU destruction rates between battalion and regimental task organizations in Scenario Two is expected; increasing the number of M777A2 batteries increases HVU destruction by 4.6%.

Table 4.1. Scenario Summary Statistics

Scenario	Artillery Unit Task Organization	HVU Destruction Rate
1	Battalion	58.9%
1	Regiment	47.7%
2	Battalion	18.6%
2	Regiment	23.3%

4.6 Partition Tree Analysis

Partition trees recursively partition data by finding relationships between predictor and response variables. Each branch, or partition, in the tree is determined by finding the predictor factor, X, and a split value for X that leads to the biggest improvement in R^2. De Veaux et al. in [16] describes R^2 as, "the fraction of the data's variation accounted for by the model". For EFSM-M a larger R^2 value indicates more variation in mean probability of HVU destruction is explained by the factors and splits in the partition tree.

4.6.1 Scenario One Analysis

Partition trees with eight partitions for battalion and regimental task organizations in Scenario One have R^2 values of 53.1% and 43.9% respectively. For both task organizations, the M777A2 effective casualty radius (ECR) emerges as the first split. Figure 4.5 shows the partition tree for Scenario One battalion task organization. Factor combinations resulting in the highest overall mean probability of HVU destruction are M777A2 effective radius greater than 60 m, UAS sensor range less than 13 NM, and M777A2 firing delay less than 28 s. The resulting mean probability of HVU destruction is 80.4%. At the three splits involving UAS capabilities, either a smaller UAS sensor range or a slower UAS speed result in a higher mean probability of HVU destruction. This finding is interesting because it is intuitive to believe a more capable UAS would increase the likelihood of detecting the HVU and, ultimately, increase the mean probability of HVU destruction. Figure 4.6 shows that UAS maximum speed, M777A2 ECR, and UAS maximum sensor range account for the majority of variation for the battalion task organization in this partition tree.

The partition tree for regimental task organization in Scenario One appears in Figure 4.7. Column contributions in Figure 4.8 show that the M777A2 firing delay is the most important. The highest mean HVU probability of destruction, as shown in Figure 4.7, is 79.6% with a combination of M777A2 ECR less than 64 m, UAS speed between 13 Knots and 14 Knots, and M777A2 firing delay less than 27 s. The regimental task organization, similar to the battalion task organization, shows combinations of smaller UAS sensor range and reduced UAS speed increase the probability of HVU destruction.

Neither partition tree for Scenario One has leaves associated with extremely low probabilities of HVU destruction. Highlighted in yellow of Figure 4.8 is a leaf with moderately low probability of HVU destruction. M777A2 munition P_k below 0.22, coupled with M777A2 firing delay of 23 s or greater, UAS maximum speed of 15 Knots or greater, and M777A2 munition ECR less than 64 m, achieves less than a 20% chance of HVU destruction.

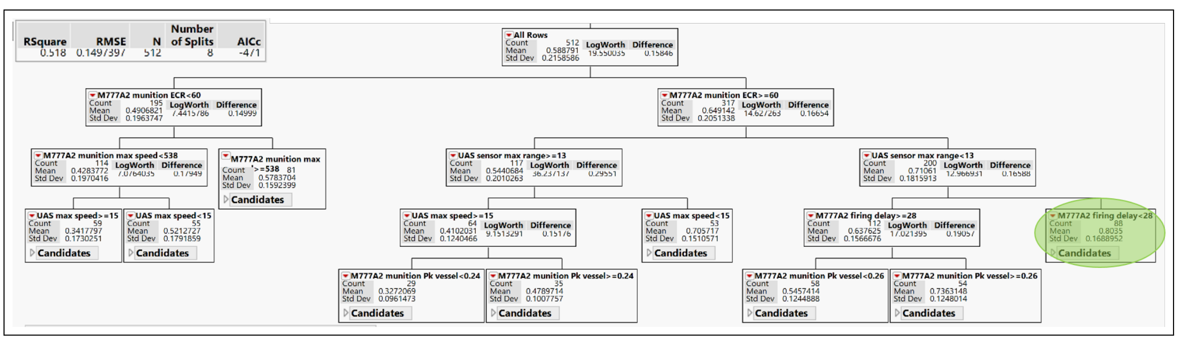


Figure 4.5. Scenario One Battalion Task Organization Partition Tree

Column Contributions

Term	Number of Splits	SS	Portion
UAS max speed	2	3.44885407	0.2797
M777A2 munition ECR	1	3.03153574	0.2459
UAS sensor max range	1	2.04739758	0.1661
M777A2 munition Pk vessel	2	1.38089411	0.1120
M777A2 firing delay	1	1.35591533	0.1100
M777A2 munition max speed	1	1.06536462	0.0864

Figure 4.6. Scenario One Battalion Task Organization Column Contributions

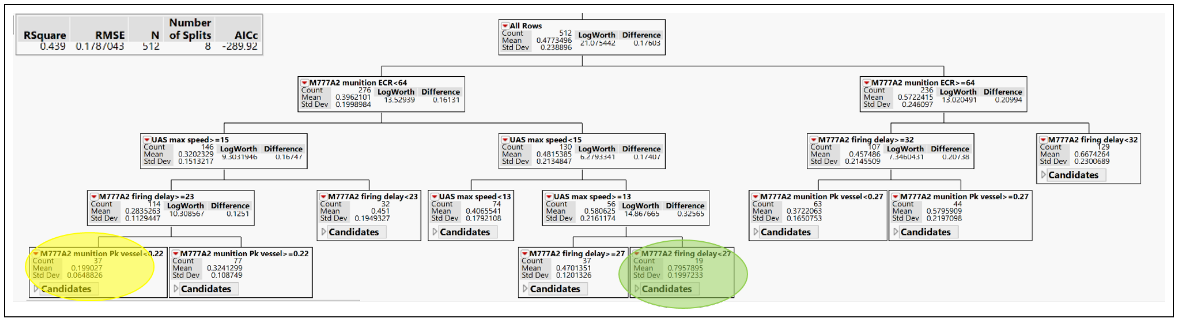


Figure 4.7. Scenario One Regiment Task Organization Partition Tree

Column Contributions

Term	Number of Splits	SS	Portion
M777A2 firing delay	3	4.60994226	0.3598
M777A2 munition ECR	1	3.94213338	0.3077
UAS max speed	2	2.75520432	0.2150
M777A2 munition Pk vessel	2	1.50532852	0.1175

Figure 4.8. Scenario One Regiment Task Organization Column Contributions

4.6.2 Scenario Two Analysis

Figure 4.9 and Figure 4.11 show the partition trees and column contributions for the two artillery task organizations for Scenario Two. Factors in Scenario Two accounts for more data variability in the response than factors in Scenario One, achieving R^2 values of 82.6% and 74%. UAS range and speed are the best predictors of HVU destruction in Scenario Two, as shown in Figure 4.10 and Figure 4.12. UAS maximum speed less than 14 Knots and UAS sensor range less than 9 NM, highlighted in green in Figure 4.9 and Figure 4.11, are examples of good leaves. Both show probability of HVU destruction above 60.3%. Leaves at the first partitions, highlighted in red, show UAS sensor range greater than 12 NM drops the chance of HVU destruction to 2.1% for the battalion task organization and 4.6% for the regimental task organization. For Scenario Two, if UAS speed is also above 13 Knots, this reduces the likelihood of HVU destruction below 1%. Leaves showing UAS speed above 13 Knots are highlighted in red for both task organizations. These are bad leaves, and demonstrate how sensitive Scenario Two is to increased UAS speed and sensor range.

Good Leaves, highlighted in green, show that sufficiently slow UASs with small sensor ranges lead to good outcomes. All the splits involving UAS speed show that reducing UAS capability increases the probability of HVU destruction. This finding also occurs in Scenario One. A possible explanation for reduced UAS sensor range and speed increasing the probability of HVU destruction is the order in which targets are engaged in EFSM-M. Targets are engaged in the order in which they are detected by sensors. UAS traveling at greater speed with increased sensor range search a larger area and may detect screen ships before the HVU. Detecting screen ships first results in less HVU engagements. This is only a hypothesis. The order of sensor detections is not an output of EFSS. Additional experimentation is required to determine the relationship between reduced UAS capability and increased probability of HVU destruction.

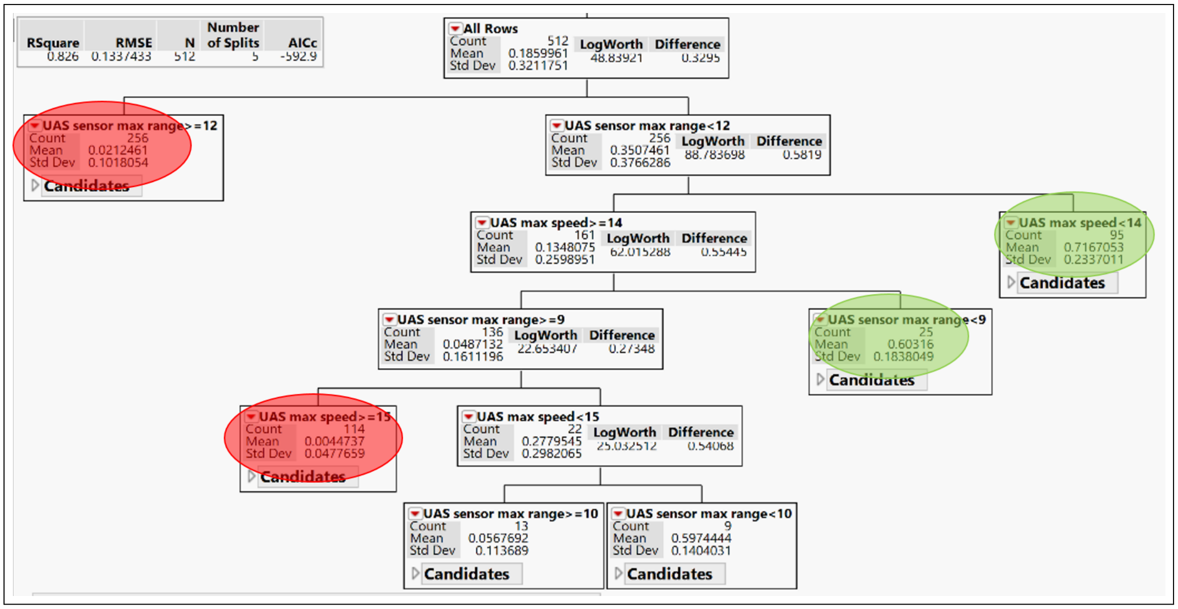


Figure 4.9. Scenario Two Battalion Task Organization Partition Tree

Column Contributions

Term	Number of Splits	SS	Portion
UAS sensor max range	3	21.9435683	0.5038
UAS max speed	2	21.6095782	0.4962
HimarsMinRange	0	0	0.0000

Figure 4.10. Scenario Two Battalion Task Organization Column Contributions

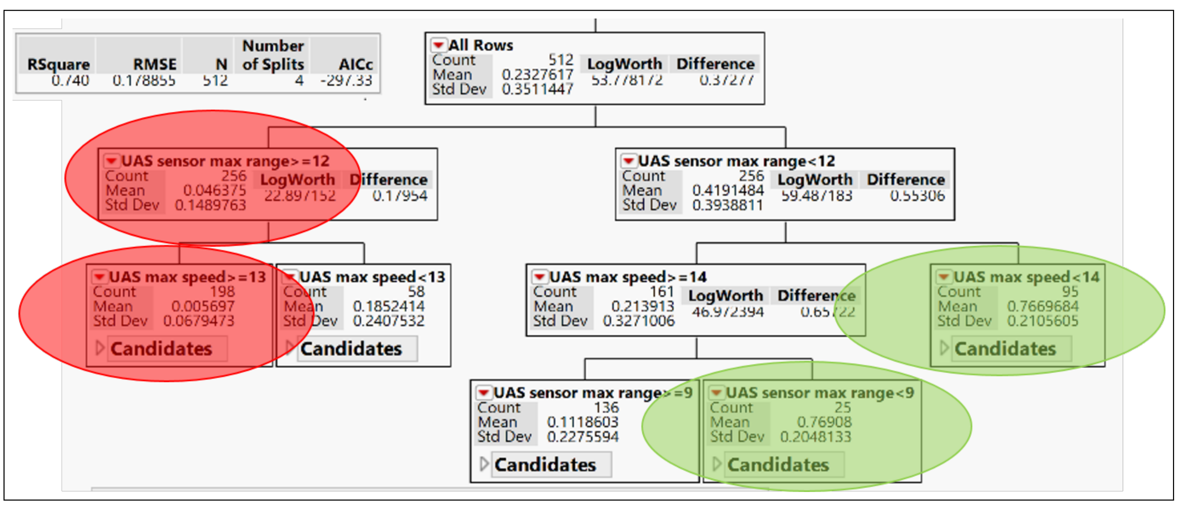
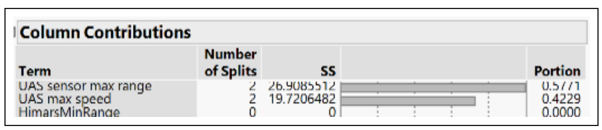


Figure 4.11. Scenario Two Regiment Task Organization Partition Tree



Column Contributions

Term	Number of Splits	SS		Portion
UAS sensor max range	2	26.9085512		0.5771
UAS max speed	2	19.7206482		0.4229
HimarsMinRange	0	0		0.0000

Figure 4.12. Scenario Two Regiment Task Organization Column Contributions

Interesting factors in Scenario One and Two are UAS maximum speed, UAS sensor maximum range, and M777A2 firing delay. Figure 4.13 and Figure 4.14 show overall trend lines for the mean probability of HVU destruction, broken down by M777A2 firing delay, versus UAS maximum speed and UAS sensor maximum range. In both figures, the differences between task organizations in Scenario Two are small. Larger differences in mean probability of HVU destruction exist between battalion and regimental task organizations in Scenario One. Both task organizations in simulation Scenario One and Scenario Two show overall downward trends in mean probability of HVU destruction as UAS speed increases from 9 Knots to 20 Knots and UAS sensor range increases from 7 NM to 15 NM. The slopes of the trend lines indicate that Scenario Two is more sensitive to increasing UAS speed than Scenario One. Decreased M777A2 firing delay has a positive affect on mean probability of HVU destruction, particularly for Scenario One.

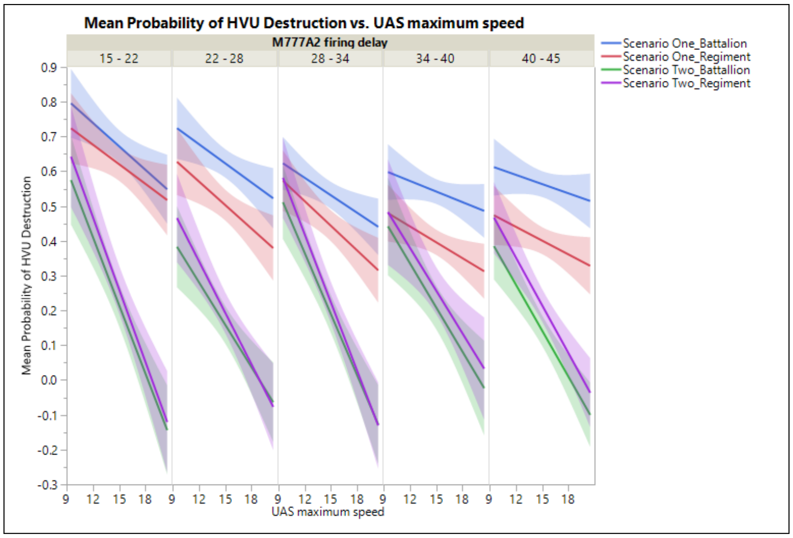


Figure 4.13. UAS Speed and M777A2 Firing Delay Average Value Trends

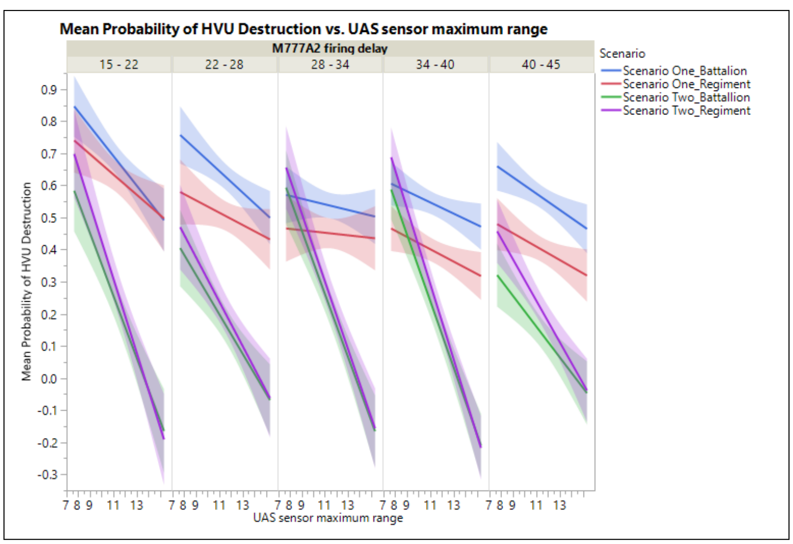


Figure 4.14. UAS Sensor Range and M777A2 Firing Delay Average Value Trends

4.7 Recommendations

Running EFSM-M using an experimental design supports decision making by quickly focusing attention on a few key factors among the 50 investigated. It also identifies interactions between factors, which is not possible without simultaneously varying factor values within the same experiment. Based on the key factors identified during analysis, and an understanding of the assumptions, possible recommendations and insights for decision makers may include:

1. Current USMC artillery systems achieve higher HVU destruction rates when convoy routes are known and vessels are limited in speed and maneuver. Commanders can best a employ artillery at choke points to limit adversary maritime maneuver.
2. Current USMC artillery systems provide limited anti-access capability to commanders. Battalion and regimental task organizations of artillery conducting anti-access operations achieved a mean HVU destruction probability of less than 25%.
3. UAS characteristics emerged as important factors in A2AD operations. Additional study is required to fully determine the importance of UASs in support of A2AD operations.
4. Neither task organization is dominant. In this study, the battalion was more effective for area denial, while the regiment was more effective for anti-access.

Analysis is not the primary purpose of this thesis. Partition trees serve to demonstrate that EFSM-M produces output data suitable for analysis from a large-scale simulation experiment. Recommendations 1–4 are examples intended to demonstrate that experiments involving EFSM-M can provide useful data to inform decision making. Recommendation 4 describes an interaction. Additional scenarios, MOEs, and analysis are required to improve recommendations and provide more insights. A wide variety of other parametric, non-parametric and graphical methods can be used for in-depth investigations. See [3] for examples of a more robust analysis of a DES model, or [17] for a general discussion.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 5: Conclusions and Future Work

The analysis and recommendations presented in this thesis only serve as a demonstration of the usefulness of EFSM-M as an analysis tool. Additional data, simulation, and model improvement is required. Recommendations for improving EFSM-M and additional scenarios for analysis are presented as future work in Section 5.2.

5.1 Conclusions

The EFSM-M, described in Chapter 3, is designed to be extensible and interoperate with other DES models. Interoperability is demonstrated by joining two models, SurfaceSim and EFSM-M, to create a littoral combat simulation. Extensibility is provided by Simkit class libraries and the ability to add functionality without altering source code. Currently, EFSM-M approximates USMC artillery and sensor capabilities including: HIMARS, EFSS, M777A2, FSTs, UAS, and FFR. Units may be created with any number of artillery system and sensor combinations, and new munitions can be added by creating new Munition-Target adjudicators. In all, EFSM-M provides analysts a capable tool for future experimentation.

EFSM-M provides decision makers a tool for exploring future system development and application by supporting experimental design. The nearly orthogonal-and-balanced, mixed experimental design used in this thesis includes 50 factors and 512 design points. Partition tree analysis indicates UAS speed and M777A2 firing rate are important factors across both scenarios. Increasing M777A2 firing rate (i.e., decreasing its firing delay) and reducing UAS speed and sensor radius can lead to higher rates of HVU destruction. The finding that reduced UAS capabilities can the chances of HVU destruction is interesting and unintuitive, and requires further investigation. Proof-of-concept analysis demonstrates the ability of EFSM-M to provide analysts and decision makers data for analyzing the effects of artillery and target locating systems in a littoral environment. Artillery performed better overall in Scenario One, the anti-access scenario. Battalion and regimental artillery task organizations achieved HVU destruction rates of 58.9% and 47.7%, respectively. Artillery did not perform as well in Scenario Two, achieving HVU destruction rates of only 18.6% and 23.3%. Given the challenges of conventional indirect fire munitions hitting a moving target at long range, low destruction rates are not unexpected. That being said, artillery is capable of providing limited offensive capability to reduce adversary maritime maneuver, but should be employed when maritime forces are limited in speed and maneuverability to maximize effectiveness.

5.2 Future Work

Future work includes continued improvement of EFSM-M functionality, scenario development, and user interface tools. Four ideas for future work are presented in this section. Additionally, using an experimental design early in the future software development process is recommended. Software testing with an experimental design revealed interoperability issues between SurfaceSim and EFSM-M that did not present during single run testing.

5.2.1 Graphical User Interface

Scenario development tools for EFSM-M should be improved. Currently, the simulation modeler is required to manually input the location of simulation entities into a Microsoft Access database with no visual feedback. The inability to visualize entities in simulation space make altering the scenario difficult and extremely time consuming. The SandboxFrame visualization tool, provided in Simkit, is helpful for initial testing, but requires the simulation modeler to compile and run the simulation after each change. Additionally, running the SandboxFrame graphical output slows the model execution and debugging processes. A graphical user interface (GUI) based scenario editor, that allows a simulation modeler to drag and drop simulation entities into simulation space, will increase the usability of EFSM-M. Reducing scenario development time will allow for more scenarios to support analysis and improve the usefulness of the model.

5.2.2 Engagement Criteria

Currently, the order targets are engaged in EFSM-M is in order of detection. Actual artillery operations, however, prioritize targets based on type of target and the relative value of that target compared to others. For the two scenarios in this thesis, an improved engagement criteria would prioritize engagement of the HVU over screen ships. In future simulations, modelers can improve the engagement criteria the FireSupportCoordinator applies when engageing targets. A better model of actual target engagement criteria is needed to fully understand the effects of artillery operating in littoral environments.

5.2.3 Artillery Unit Location

EFSM-M does not dynamically place artillery units and sensors in simulation space. In military operations, artillery units are continuously moving to support ground forces and increase their own survivability by taking advantage of weapon standoff range. To more accurately approximate artillery operations, EFSM-M should dynamically place artillery units based on weapon system range and a designated target area. A search algorithm that identifies firing positions, based on the minimum and maximum range of the artillery munitions and an aim point, will better approximate artillery operations and adds artillery survivability as a possible MOE for analysis.

5.2.4 Unmanned Systems Behavior

UAS in EFSM-M fly between user-defined waypoints and never alter course. A more realistic representation of UAS is tracking targets once an initial detection event occurs and the detected entity is identified as a target of interest. To accomplish this, a path finding algorithm that updates based on the track of another simulation entity may be incorporated to better approximate the operations of UAS. Sensors actively tracking targets will also require an update to the engagement algorithm of the FireSupportCoordinator class. Engagement criteria will need to include when to engage tracked targets and when to order a UAS to return to its original path.

5.2.5 Scenario Development

The simulation scenarios developed for this thesis are a proof-of-concept to demonstrate model functionality. A fully developed littoral combat scenario is required to better inform decision makers on the capabilities of current USMC artillery capabilities conducting A2AD operations. Suggestions for future scenarios include examining A2AD operations using defense in depth, adding merchant ship traffic to scenarios, and examining the effect of artillery munitions capable of actively tracking moving targets.

5.3 Closing Thoughts

Today, many nations are able to challenge for control of the sea because powerful, blue water navies are no longer a prerequisite for maritime power. ASMs and small boat tactics have leveled the field allowing non-state actors and smaller nations to influence world trade by focusing efforts at strategic maritime choke points. EFSM-M provides the Navy and Marine Corps an extensible tool to study current weapon capabilities and operating concepts to ensure they stand ready for the future littoral fight.

APPENDIX: Factor and Parameter Descriptions and Values

Table A.1. EFSM-M Factor Descriptions and Ranges

Number	Factor description	Minimum value	Maximum value	Details
1	HIMARS firing delay	10 s	20 s	Time between rounds fired
2	HIMARS munition ECR	50 m	75 m	The effective casualty radius of a circle impacted by the detonation of a HIMARS munition
3	M777A2 munition ECR	50 m	75 m	The effective casualty radius of a circle impacted by the detonation of an M777A2 munition
4	EFSS munition ECR	60 m	80 m	The effective casualty radius of a circle impacted by the detonation of an EFSS munition
5	HIMARS munition maximum speed	950 m/s	1050 m/s	The maximum speed a HIMARS can travel
6	M777A2 munition maximum speed	450 m/s	600 m/s	The maximum speed an M777A2 can travel
7	EFSS munition maximum speed	275 m/s	325 m/s	The maximum speed an EFSS can travel

Number	Factor description	Minimum value	Maximum value	Details
8	HIMARS munition P_k vessel	0.15	0.35	The probability a single HIMARS munition impact destroys a vessel
9	M777A2 munition P_k vessel	0.15	0.35	The probability a single M777A2 munition impact destroys a vessel
10	EFSS munition P_k vessel	0.15	0.35	The probability a single EFSS munition impact destroys a vessel
11	HIMARS munition maximum range	80000 m	120000 m	The maximum range of a HIMARS munition
12	M777A2 munition maximum range	18000 m	33000 m	The maximum range of an M777A2 munition
13	EFSS munition maximum range	1000 m	7600 m	The maximum range of an EFSS munition
14	HIMARS munition minimum range	0 m	5000 m	The minimum range of a HIMARS munition
15	M777A2 munition minimum range	0 m	0 m	The minimum range of an M777A2 munition
16	EFSS munition minimum range	0 m	600 m	The minimum range of an EFSS munition
17	Artillery unit do kill efficiency	70%	90%	The threshold value to kill an artillery unit

Number	Factor description	Minimum value	Maximum value	Details
18	Artillery unit gun dispersion	50 m	150 m	The distance between artillery pieces
19	Artillery unit mission process time	45 s	75 s	The time it takes an artillery unit to calculate firing data
20	Fire support team mean sensor distortion along X axis	75 m	100 m	TLE measured from the target along the sensor-target line
21	Fire support team sensor distortion along Y axis	30 m	40 m	TLE measured perpendicular to the sensor-target line
22	Fire support team sensor distortion along X axis standard deviation	0 m	5 m	Standard deviation of sensor distortion along the sensor-target line
23	Fire support team sensor distortion along Y axis standard deviation	0 m	5 m	Standard deviation of sensor distortion perpendicular to the sensor-target line
24	Fire support team sensor maximum range	4500 m	6500 m	The maximum range of FST sensor
25	UAS sensor distortion along X axis	5 m	15 m	TLE measured from the target along the sensor-target line
26	UAS sensor distortion along Y axis	5 m	15 m	TLE measured perpendicular to the sensor-target line

Number	Factor description	Minimum value	Maximum value	Details
27	UAS sensor distortion along X axis standard deviation	0 m	4 m	Standard deviation of sensor distortion along the sensor-target line
28	UAS sensor distortion along Y axis standard deviation	0 m	5 m	Standard deviation of sensor distortion perpendicular to the sensor-target line
29	UAS sensor maximum range	8 NM	15 NM	The maximum range of the UAS sensor
30	UAS maximum speed	10 knots	20 knots	The maximum speed of the UAS
31	Fire finder radar sensor distortion along X axis	5 m	15 m	TLE measured from the target along the sensor-target line
32	Fire finder radar sensor distortion along Y axis	10 m	20 m	TLE measured perpendicular to the sensor-target line
33	Fire finder radar sensor distortion along X axis standard deviation	0 m	5 m	Standard deviation of sensor distortion along the sensor-target line
34	Fire finder radar sensor distortion along Y axis standard deviation	0 m	5 m	Standard deviation of sensor distortion perpendicular to the sensor-target line
35	Fire finder radar sensor maximum range	80000 m	100000 m	The maximum range of the fire finder radar sensor

Number	Factor description	Minimum value	Maximum value	Details
36	Fire finder radar time to detect	3 s	5 s	The time a round must be within the sensor's range before it is detected
37	Fire finder radar probability of detection	0.8	0.95	Probability the fire finder radar correctly identifies the point of origin of an incoming munition
38	Number of HIMARS per artillery unit	2	4	Integer number of individual HIMARS in a firing battery
39	Number of M777A2 per artillery unit	1	9	Discrete (3 levels) number of individual M777A2 in a firing battery
40	Number of EFSS per artillery unit	3	6	Discrete (2 levels) number of individual EFSS in a firing battery
41	Number of volleys	1	5	The number of artillery volleys an artillery unit fires at a target
42	Vessel maximum speed	2 Knots	10 Knots	The maximum speed of the naval convoy
43	M777A2 firing delay	15 s	45 s	Time between rounds fired
44	EFSS firing delay	10 s	25 s	Time between rounds fired

Number	Factor description	Minimum value	Maximum value	Details
45	HIMARS munition minimum damage value	0.1	0.2	Minimum input value to damage function for HIMARS munition
46	HIMARS munition maximum damage value	0.3	0.5	Maximum input value to damage function for HIMARS munition
47	M777A2 munition minimum damage value	0.1	0.2	Minimum input value to damage function for M777A2 munition
48	M777A2 munition maximum damage value	0.3	0.5	Maximum input value to damage function for M777A2 munition
49	EFSS munition minimum damage value	0.1	0.2	Minimum input value to damage function for EFSS munition
50	EFSS munition maximum damage value	0.3	0.5	Maximum input value to damage function for EFSS munition

Table A.2. SurfaceSim Parameter Descriptions and Values

Number	Parameter description	Parameter value	Details
1	Vessel policy	Shoot look shoot	Specifies a vessels shoots, checks the target, then shoots again
2	Number of times to shoot before look	1	Number of SAMs to fire before looking
3	Vessel total number of ASMs	0	Total number of ASMs per vessel (ASM engagements not included in EFSM-M simulation)
4	Vessel total number of SAMs	0	Total number of SAMs per vessel (SAM engagements not included in EFSM-M simulation)
5	Vessel firing rate	15 s	Time between vessel gun rounds fired
6	Vessel close-in weapon system (CIWS) firing rate	5 s	Time between vessel CIWS rounds fired (CIWS engagements not included in EFSM-M simulation)
7	Vessel classification time	15 s	Time require to classify an enemy unit
8	Vessel detection rate for surveillance sensors	5	Number of targets classified per minute by a sensor

Number	Parameter description	Parameter value	Details
9	Vessel ASM launch delay	20 s	Time between target classification and ASM launch
10	Vessel SAM launch delay	10 s	Time between target classification and SAM launch
11	ASM range	0 NM	Maximum range of ASM
12	SAM range	0 NM	Maximum range of SAM
13	Vessel gun range	13.5 NM	Maximum range of vessel gun
14	Vessel CIWS range	0 NM	Maximum range of vessel CIWS
15	ASM damage probability	0	Probability one ASM damages a vessel (vessel-on-vessel engagements not included in EFSM-M simulation)
16	Vessel gun round vessel damage probability	0	Probability one gun round damages a vessel (vessel-on-vessel engagements not included in EFSM-M simulation)
17	Vessel gun round FST damage probability	0.8	Probability one gun round damages a FST

Number	Parameter description	Parameter value	Details
18	Vessel gun round artillery unit damage probability	0.8	Probability one gun round damages an artillery unit
19	Vessel gun round artillery damage probability	0.8	Probability one gun round damages an individual artillery system
20	Vessel gun round ASM kill probability	0	Probability one gun round kills an ASM (gun round–ASM engagements not included in EFSM-M simulation)
21	SAM ASM kill probability	0	Probability a SAM kills an ASM (SAM–ASM engagements not included in EFSM-M simulation)
22	CIWS ASM kill probability	0	Probability a CIWS round kills an ASM (CIWS–ASM engagements not included in EFSM-M simulation)
23	ASM close enough distance for impact	0 m	The distance required for an ASM to impact
24	SAM close enough distance for impact	0 m	The distance required for an SAM to impact

Number	Parameter description	Parameter value	Details
25	Vessel gun round close enough distance for impact	0 m	The distance required for a vessel gun round to impact
26	Mean surveillance sensor distortion on x axis	0 m	Mean of rotated bivariate normal distribution on x axis for surveillance sensor
27	Mean surveillance sensor distortion on y axis	0 m	Mean of rotated bivariate normal distribution on y axis for surveillance sensor
28	Standard deviation of surveillance sensor distortion on x axis	0 m	Standard deviation of rotated bivariate normal distribution on x axis for surveillance sensor
29	Standard deviation of surveillance sensor distortion on y axis	0 m	Standard deviation of rotated bivariate normal distribution on y axis for surveillance sensor
30	Mean engagement sensor distortion on x axis	0 m	Mean of rotated bivariate normal distribution on x axis for engagement sensor
31	Mean engagement sensor distortion on y axis	0 m	Mean of rotated bivariate normal distribution on y axis for engagement sensor

Number	Parameter description	Parameter value	Details
32	Standard deviation of engagement sensor distortion on x axis	0 m	Standard deviation of rotated bivariate normal distribution on x axis for engagement sensor
33	Standard deviation of engagement sensor distortion on y axis	0 m	Standard deviation of rotated bivariate normal distribution on y axis for engagement sensor
34	ASM minimum damage	0%	Minimum damage an ASM inflicts on a target
35	ASM maximum damage	0%	Maximum damage an ASM inflicts on a target
36	Vessel gun round minimum damage	1%	Minimum damage a vessel gun round inflicts on a target
37	Vessel gun round maximum damage	2%	Maximum damage a vessel gun round inflicts on a target
38	ASM maximum speed	2000 Knots	Maximum speed of an ASM
39	SAM maximum speed	2500 Knots	Maximum speed of a SAM
40	Vessel gun round maximum speed	700 Knots	Maximum speed of a vessel gun round
41	CIWS round maximum speed	2000 Knots	Maximum speed of a CIWS round

Number	Parameter description	Parameter value	Details
42	Vessel surveillance sensor time delay until detection	5 s	Time required for surveillance sensor to detect a target
43	Vessel efficiency threshold	85%	Threshold value required to kill a vessel
44	HVU CIWS range	0 NM	Maximum range of CIWS
45	HVU CIWS close enough distance for ASM impact	0 m	The distance required for the HVU CIWS round to impact an ASM
46	HVU CIWS inter-shoot delay	5 s	Time between HVU CIWS rounds
47	ASM mean damage	0%	Mean damage an ASM inflicts on a target
48	SAM mean damage	0%	Mean damage an SAM inflicts on a target
49	Vessel gun round mean damage	1.5%	Mean damage a vessel gun round inflicts on a target
50	ASM damage standard deviation	0%	Damage standard deviation for an ASM
51	SAM damage standard deviation	0%	Damage standard deviation for a SAM
52	Vessel gun round damage	0.5%	Damage standard deviation for a vessel gun round

Number	Parameter description	Parameter value	Details
53	Vessel target check interval	15 s	Interval between check target events

THIS PAGE INTENTIONALLY LEFT BLANK

List of References

[1] T. S. Joergensen, "U.S. Navy operations in littoral waters: 2000 and beyond," *Naval War College Review*, vol. 51, no. 2, pp. 20–29, 1998.

[2] J. Kraska, "Sovereignty at sea," *Survival*, vol. 51, no. 3, pp. 13–18, 2009.

[3] A. E. Opcin, "Modeling anti-air warfare with discrete event simulation and analyzing naval convoy operations," M.S. thesis, Naval Postgraduate School, Monterey, CA, 2016.

[4] J. C. Schulte, "An analysis of the historical effectiveness of anti-ship cruise missiles in littoral warfare," M.S. thesis, Naval Postgraduate School, Monterey, CA, 1994.

[5] W. P. Hughes, "A salvo model of warships in missile combat used to evaluate their staying power," *Naval Research Logistics*, vol. 42, pp. 267–289, 1995.

[6] M. J. Armstrong, "A stochastic salvo model for naval surface combat," *Operations Research*, vol. 53, no. 5, pp. 830–841, Sep. - Oct. 2005.

[7] C. M. Mahon, "A littoral combat model for land-sea missile engagement," M.S. thesis, Naval Postgraduate School, Monterey, CA, 2007.

[8] L. Schruben, "Simulation modeling with event graphs," *Commun. ACM*, vol. 26, no. 11, pp. 957–963, Nov. 1983.

[9] A. H. Buss, "Discrete event simulation modeling," 2016, unpublished.

[10] A. H. Buss, "Component based simulation modeling with Simkit," in *Proc. 2002 Winter Simulation Conf.*, 2002, pp. 243–249.

[11] A. H. Buss, "Discrete event programming with Simkit," *Simulation News Europe*, Nov. 2001.

[12] A. H. Buss and P. J. Sanchez, "Simple movement and detection in discrete event simulation," in *Proc. 2005 Winter Simulation Conf.*, 2005, pp. 992–1000.

[13] The Java Tutorials Enum Types. (n.d). Oracle. [Online]. Available: https://docs.oracle.com/javase/tutorial/java/javaOO/enum.html. Accessed Apr. 25, 2017.

[14] H. Vieira Jr., S. Sanchez, K. Kienitz, and M. Belderrain, "Efficient, nearly orthogonal-and-balanced, mixed designs: an effective way to conduct trade-off analyses via simulation," *J. of Simulation*, no. 7, pp. 264–275, 2013.

[15] T. M. Cioppa and T. W. Lucas, “Efficient nearly orthogonal and space-filling Latin hypercubes,” *Technometrics*, vol. 49, no. 1, pp. 45–55, Feb. 2007.

[16] R. D. DeVeaux, P. F. Velleman, and D. E. Bock, *Intro Stats*, 3rd ed. Boston: Pearson/Addison-Wesley, 2009.

[17] S. M. Sanchez, “Simulation experiments: Better data, not just big data,” in *Proc. of the 2015 Winter Simulation Conf.*, Piscataway, NJ, 2015, pp. 800–811.

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California